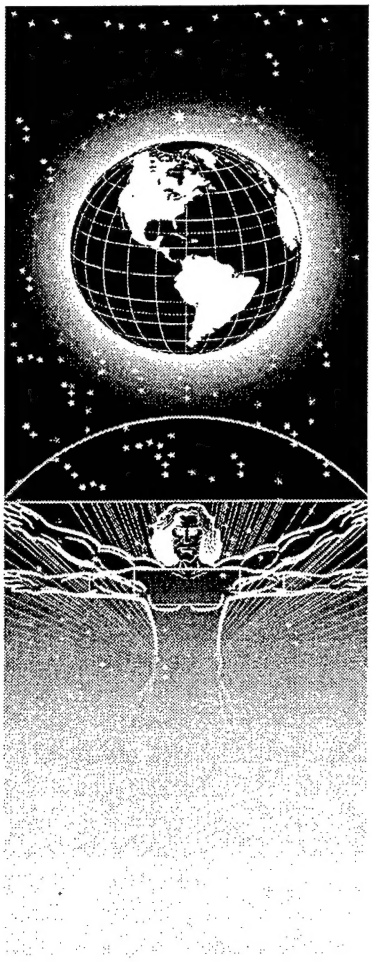
AFRL-HE-WP-TR-2001-0064



UNITED STATES AIR FORCE RESEARCH LABORATORY

Feasibility of Direct Conversion of Page-Based Information

Sherrise L. McClellan
Nicholas J. Stute

TASC, Inc.
2555 University Blvd.
Fairborn, OH 45324

Colleen Woolley

TAMSCO
3070 Presidential Drive, Suite 300
Fairborn, OH 45324

Maurice C. Azar

Air Force Research Laboratory

March 1998

Final Report for the Period July 1997 to March 1998

Human Effectiveness Directorate
Deployment and Sustainment Division
Logistics Readiness Branch
2698 G Street
Wright-Patterson AFB OH 45433-7604

NOTICES

When US Government drawings, specifications or other data are used for any purpose other than a definitely related Government procurement operation, the Government thereby incurs no responsibility nor any obligation whatsoever, and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications or other data, is not to be regarded by implication or otherwise, as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

Please do not request copies of this report from the Air Force Research Laboratory. Additional copies may be purchased from:

National Technical Information Service
5285 Port Royal Road
Springfield, VA 22161

Federal Government agencies registered with the Defense Technical Information Center should direct requests for copies of this report to:

Defense Technical Information Center
8725 John J. Kingman Rd., Ste 0944
Ft. Belvoir, VA 22060-6218

DISCLAIMER

This Technical Report is published as received and has not been edited by the Air Force Research Laboratory, Human Effectiveness Directorate.

TECHNICAL REVIEW AND APPROVAL

AFRL-HE-WP-TR-2001-0064

This report has been reviewed by the Office of Public Affairs (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER

ALBERT S. TORIGIAN, Lt Col, USAF
Deputy Chief
Deployment and Sustainment Division
Air Force Research Laboratory

REPORT DOCUMENTATION PAGE	Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	March 1998	Final - July 1997 - March 1998

4. TITLE AND SUBTITLE
Feasibility of Direct Conversion of Page-Based Information

5. FUNDING NUMBERS
C: F41624-97-D-5002
PE: 63106F
PR: 2950
TA: 00
WU: 45

6. AUTHOR(S)
*Sherrise L. McClellan, *Nicholas J. Stute, **Colleen Woolley,
Maurice C. Azar

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
*TASC, Inc.
2555 University Blvd.
Fairborn, OH 45324

** TAMSCO
3070 Presidential Drive, Suite 300
Fairborn, OH 45324

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Research Laboratory, Human Effectiveness Directorate
Deployment and Sustainment Division
Air Force Materiel Command
Logistics Readiness Branch
Wright-Patterson AFB, OH 45433-7604

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
AFRL-HE-WP-TR-2001-0064

11. SUPPLEMENTARY NOTES
AFRL Monitor: Cheryl L. Batchelor, AFRL/HESR, 937-656-4392

12a. DISTRIBUTION AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

Feasibility of Direct Conversion of Page-Based Information is a study to determine the feasibility of using low cost Commercial Off-the-Shelf (COTS) software packages to convert paper based technical information to HyperText Markup Language (HTML) and viewed via low cost web browsers. As well as determining feasibility, the project evaluated multiple methods of conversion with alternate electronic file formats (i.e., Standard Generalized Markup Language (SGML) and Portable Document Format (PDF). Each method was investigated and a recommended process to convert paper technical manuals to HTML using specific COTS software packages is presented. Costs to convert the paper to HTML using this process were quantified and compared to costs for electronic formats (Interactive Electronic Technical Manuals (IETMs)). A demonstration of the resulting HTML files was conducted and is described in this final report.

14. SUBJECT TERMS
HyperText Markup Language (HTML) Standard Generalized Markup Language (SGML)
Commercial Off-the-Shelf (COTS) Software Optical Character Recognition (OCR)
IETMs Scanning Hyperlink Web Browser Conversion

15. NUMBER OF PAGES
84

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

THIS PAGE LEFT INTENTIONALLY BLANK

PREFACE

This work was funded under Work Unit 2950-00-45, D.O. 009. Special thanks go to 1LT. Maurice Azar of the Crew Survivability and Logistics Division, Logistics Readiness Branch for implementing and managing this project.

TABLE OF CONTENTS

Page

TABLE OF CONTENTS (Continued)

LIST OF FIGURES

LIST OF TABLES

LIST OF TABLES (Continued)

1.0 INTRODUCTION

At the United States Air Force (USAF) Air Force Research Laboratories/Crew Survivability and Logistics Division, Logistics Readiness Branch (AFRL/HESR) request, TASC was contracted to perform a study of the feasibility of direct conversion of page-based information to a HyperText Markup Language (HTML) 3.2 format. The overall goal of this project was to identify a quick, cost effective process to convert paper data to a digital format supported by web browsers using Commercial Off The Shelf (COTS) software and to reduce the time required for a maintenance technician to retrieve the information required to perform specific aircraft maintenance. Toward that goal, the objectives of this task were to:

- Quantify and validate the benefits of using different methods to convert paper technical data to electronic format for use by USAF maintenance technicians using portable computer hardware and software

- Provide an example set of text (per World Wide Web Consortium (WC3) HTML 3.2) and graphic material for use by either of the two mentioned web browsers (Netscape Navigator and/or Microsoft (MS) Internet Explorer)

- Document the process used to go from page-based documents to a digital data set useable by commercial web products

Based on the desire to not create custom software, this feasibility study researched commercially available products that support converting paper source documents into electronic data for viewing via commercially available web-based browsers.

This study included the presentation of identified methods and preliminary cost data, interaction with a government support organization to allow human factors inputs, a final presentation of the conversion process using an example data set in a one day open forum presentation/demonstration format, and this final written report. Additionally, a method for configuration control was recommended for a full implementation.

Toward the goal of this study, the TASC/Technical and Management Services Corporation (TAMSCO) team members investigated technical information source material, business environment, and current technology available.

Primary Conversion of Paper-Based Information Study (COPIS) team members were Ms. Sherri McClellan and Mr. Nick Stute of TASC, Inc. and Ms. Colleen Woolley and Mr. Troy Pearson of TAMSCO.

2.0 EXECUTIVE SUMMARY

The purpose of the project, COPIS, was to develop an efficient process for the conversion of page-based technical order material to a digital format supported by commercially available web-based software products (such as MS Internet Explorer or Netscape's Navigator/ Communicator) and accessed via industry standard portable computers.

Technical order (TO) information usually consists of text and graphical illustrations that define the manufacturer's recommended maintenance procedures for a specific system or system component. Currently, the U.S. Department of Defense (DOD) has sponsored several technology initiatives to facilitate the shift of this paper information to electronic formats and electronic document accessibility. Thus far, these initiatives have produced methods that are expensive to conduct, proprietary in nature, and in some cases difficult to utilize.

The COPIS team has gained an understanding of the current environment, researched current DOD conversion methods, researched and defined various conversion technologies and COTS software, and performed tests of paper TO samples converted to usable HTML files.

While evaluating and testing the conversion processes and software packages, the COPIS team defined a process to convert paper technical information to HTML. This process includes Caere WordScan Plus 4.0 (for converting paper to an MS Word file), manual intervention for formatting, InfoAccess HTML Transit 3.0 (for translation of the MS Word file to HTML 3.2 with hyperlinks), and manual insertion of hyperlinks to separate but related HTML documents. A second process was defined in order to convert SGML (Standard Generalized Markup Language) to HTML. The HTML outputs from both outputs were compared as well as the costs associated with the conversions. These costs were also compared with costs associated with the current DOD Interactive Electronic Technical Manual (IETM) electronic format. The COPIS team also received inputs from government human factors personnel to ensure the usability of the HTML files. Recommendations for a configuration management process for managing and distributing updates of the technical information was developed as a result of COPIS.

Some of the challenges the team encountered during the study were the proprietary nature of historical electronic technical data, the quality of original paper TOs, the sophistication of low cost commercial conversion tools, and the rapid emergence of new conversion technologies, electronic formats (Adobe's Portable Document Format (PDF) proprietary headers, HTML versions, and the future acceptance of Extensible Markup Language (XML)), and COTS software package upgrades.

COPIS has demonstrated:

- The feasibility of converting paper-based technical information to HTML using inexpensive COTS software packages
- That HTML hyperlink capabilities provide for ease of use while information is being retrieved and viewed through a commercially available web browser

- That costs associated with the paper to HTML conversion process using current technology in inexpensive COTS software packages are significantly less than the costs of converting paper to IETMS at the time of this study

- Technology available for the paper to HTML conversion process still requires significant manual intervention. (During COPIS, four of the seven software packages thoroughly evaluated introduced major upgrades. Each upgrade incorporated improvements to the quality of the output leading to decreases in manual intervention.)

3.0 METHODS, ASSUMPTIONS, AND PROCEDURES

The first portion of this paper discusses the current use of technical manuals in the maintenance environment. Understanding the typical maintenance manual distribution scenario, technical manuals, and current digital formats already in use by the DOD will assist in narrowing available techniques to convert paper to usable electronic formats. As a baseline for the study, information is included on a current electronic format, IETMs. This information provides the base line set of parameters in order to measure the proposed web-based process.

A sample of paper TOs that had previously been converted from paper to IETMs was gathered for document analysis. This sampling of TOs was assumed to be representative of paper technical information. Successful conversions included text, graphics, tables, and large format graphics (11" x 17" size).

As part of a technology assessment to determine the various conversion paths available for study, TASC and TAMSCO participated in several meetings with 1LT. Azar (AFRL/HESR Program Manager) to identify requirements and to assess and exchange information regarding our knowledge in the relevant areas. Combined areas of knowledge include electronic authoring tools for TO information, current DOD IETMs, and our experiences with paper conversion to electronic formats such as MS Word file (DOC), Rich Text Format (RTF), SGML, HTML, and Adobe's PDF. These information exchanges precipitated discussion regarding the different HTML formats, the Air Force Product Data Systems Modernization's (AF PDSM) conversion of paper data to PDF format, and the introduction of XML format.

Based on current efforts, such as the AF PDSM Technical Order Conversion Operation (conversion of paper data to PDF), current electronic file types used in the DOD (SGML and IETM), and the feasible formats for COTS software (MS Word and HTML), the following conversion methods were evaluated:

- Paper to IETMS (MS Word to SGML and SGML to IETMS)
- Paper to HTML (Paper scanned and converted to MS Word and MS Word to HTML)
- MS Word (authored) to HTML
- SGML to HTML
- Paper to PDF format.

To identify COTS software packages available, which were related to the selected conversions, research was conducted through the use of the Internet, Computer Select (software product and periodical CD-ROM resource), miscellaneous publications, and experiences of members of the COPIS team on other related efforts. Identified software packages were evaluated for price, manufacturer stability and market recognition, and perceived industry reliability. Total budget to purchase software for COPIS was $2,000. While remaining within

the cost levels of this budget, numerous possibilities arose. After a careful sorting of the initial possibilities based on platform, software package features, description, and functionality the choices were narrowed further. Ten packages were evaluated for available software features and quality of various electronic outputs. Based on the results, evaluation of six of the ten software packages continued through the testing of source documents using weighted criteria and numeric values.

Additional assumptions made for the purpose of this study are:

- The Office of Primary Responsibility (OPR) will still distribute updates to the Technical Order Distribution Office (TODO).
- The OPR will use a Web server for distribution.
- MS Windows NT or 95 operating system is resident on portable (laptop) personal computer (PC) at the maintenance site with commercial Web browser software installed (HTML view format).
- Maintainers have basic aptitude to use or have previous experience using MS Windows based software.
- Electronic TOs will be delivered on CD-ROM.

4.0 CURRENT MAINTENANCE ENVIRONMENT

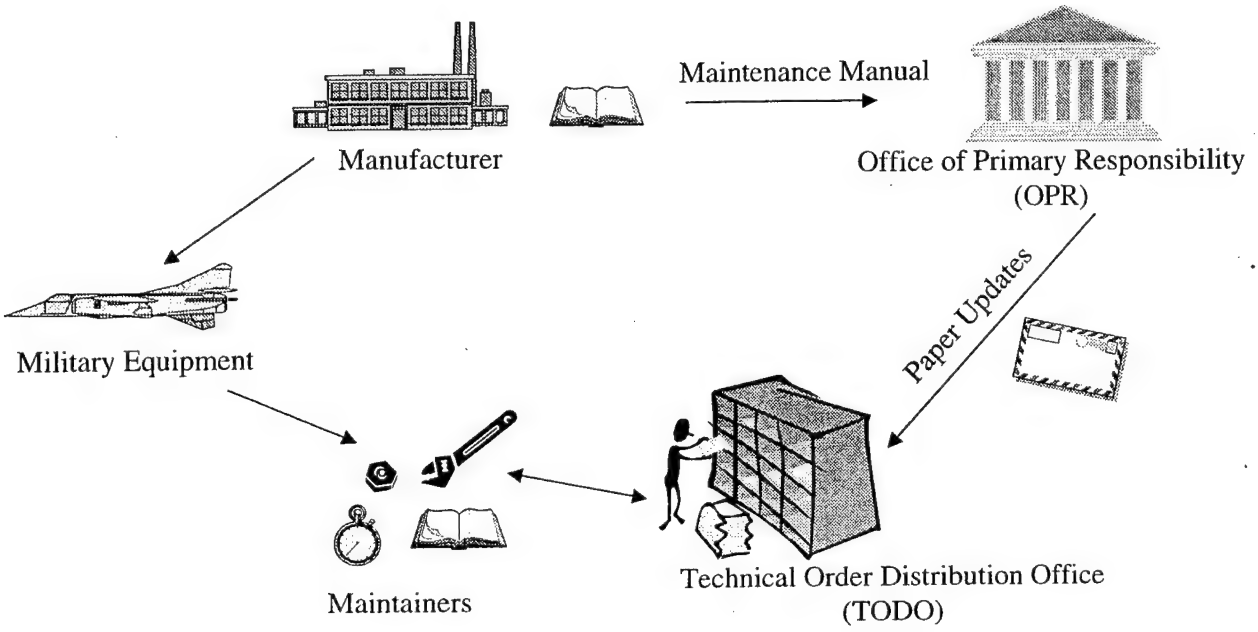


Figure 4-1. Current TO Management and Retrieval Environment

Technical manuals on the flightline are referred to as TOs. The following describes how technicians obtain paper based TO information. It is troublesome for technicians to find information within a TO. Most documents are very large; and to find the correct information, the technician may have to search page by page within a chapter or task.

In the page-based environment, the authority for a TO is assigned to an OPR. This Office is responsible for the distribution and tracking of the information. When an update or edit to a manual occurs, the OPR sends paper-based replacement pages of the manual to sites that own a copy of the respective manual in their technical maintenance library. Each site that owns a library must replace the updates by hand for each page changed. Disadvantages include lost or misplaced pages and time involved in manual updates at the TODO.

Figure 4-2 depicts the flow of technical manuals from the paper and electronic versions to the electronic outputs viewable for performing manufacturers' maintenance procedures evaluated in this study.

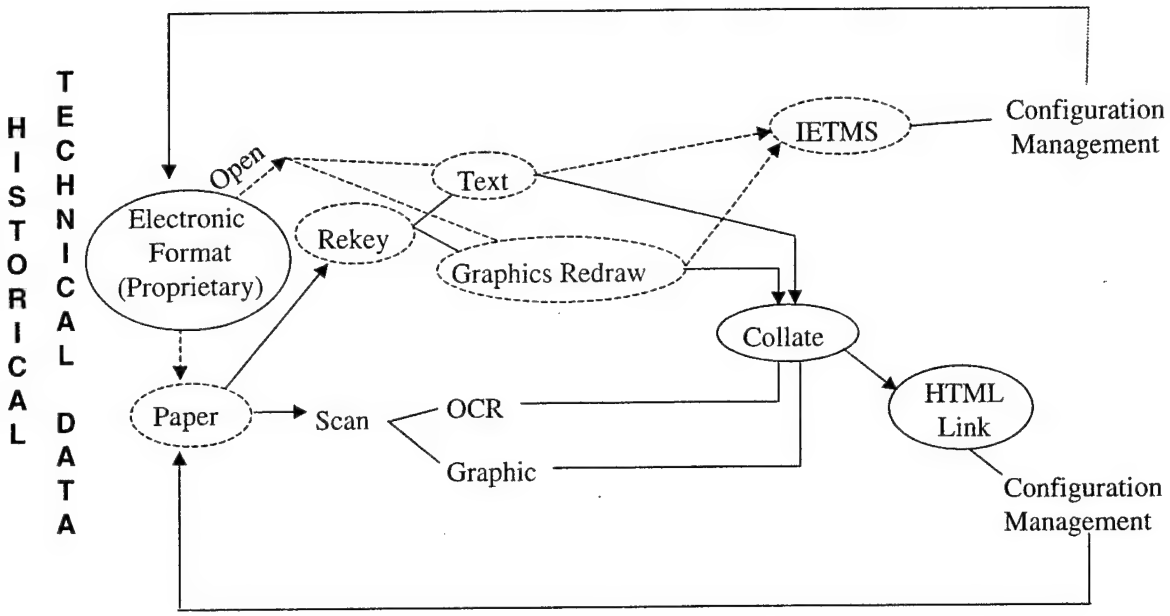


Figure 4-2. Flow of Technical Manuals to Electronic Output

Historical technical data represents TOs in original format as they were authored. It is assumed that each manufacturer has a different way of producing TOs. In many instances, the TOs reside with the manufacturer in an electronic format that is unavailable to the DOD. (The Statement of Work provided for paper information.) In some cases, even if the manufacturer would provide or sell the electronic version of the TO, the format is proprietary and the cost to convert the legacy data to an open electronic format is prohibitive. Therefore, these TOs are regarded as paper format as if the electronic format never existed.

The conversion paths for paper are depicted in Figure 4-2. Paper information can be re-keyed or scanned. In either instance, text and graphics are separated and then collated for the electronic document. Regardless of the format, HTML (with graphic files) or IETM, the information within the electronic TO needs to reference additional supporting technical information found within and between TOs (linking). Additionally, configuration management is instrumented for the management of updates and changes to TOs.

If a non-proprietary format is available to the DOD, the graphics are separated from the text and converted to HTML (with graphic files) or IETM. (Some contracts have made provisions for the manufacturer to provide an electronic version of the TO.)

5.0 DOCUMENT ANALYSIS

The conversion methods tested have been measured against a current on-going effort that is delivering MIL-PRF-87269 compliant TOs. This analysis of the current effort provided a baseline to fairly compare costs of current DOD IETM processes to the choices investigated and tested in the study. A 300 page representative sample of TO information comprised of types of information found in a typical USAF TO has been defined and gathered. These pages include text, graphics, and tables. This sample of TOs have been previously converted from MS Word DOC format to an IETM format. The MS Word files had been authored in the style and format of the USAF Military Specification MIL-M-83495, and had also been SGML tagged in accordance with MIL-M-83495 legacy Document Type Definitions (DTD). Since electronic copies of the TOs had previously been converted to IETMs, the COPIS team was able to access copies of paper TOs from TAMSCO's Technical Library. Therefore the technical data sampling consists of MS Word files (electronic), original laser printed technical data (clean paper), and degenerated paper copies (dirty paper), all which are representative of data currently used today.

Table 5-1 details specific TO pages converted as the baseline case.

Table 5-1. TOs Previously Converted to IETMs

Title	Pages Converted
TO 1C-130(A)U-2-52JG-40-1	Entire document (30 pages)
TO 1C-130(A)U-2-21JG-90-1	Entire document (128 pages)
TO 1C-130(A)U-2-29JG-00-1-1	Entire document (133 pages)
TO 1C-130(A)U-2-21GS-00-1	Pages vi through x, paragraph 1.9, and pages 11-1 through 11-4 (approximately 9 pages)
TO 1C-130(A)U-2-29GS-00-1	Only the paragraphs pertaining to the AC-130U on the pages v through ix, and chapters 1 through 4 (approximately 17 pages)
TO 1C-130(A)U-2-29JG-00-1-2	Pages 8-79 through 8-106 (approximately 27 pages)
TO 1C-130(A)U-2-00GE-00-1	Only the paragraphs applicable to the AC-130U general, air conditioning, or hydraulic systems on pages x through xiii, 1-6, 1-7, 2-1 through 3-99, 4-1 through 4-3, 5-1 through 5-8, 5-13, 5-21, 12-1, and 12-4 (approximately 15 pages)

The overall quality of the paper documents determines the types of technology necessary to complete a successful conversion. Document quality determines the amount of effort needed to prepare paper documents for an electronic conversion. Page quality is directly related to image quality. Page quality is also related to text output for Optical Character Recognition (OCR). The page qualities are described below. It is estimated that original laser print (clean pages) will OCR process with a high quality output (96% or higher image to text output accuracy). The dirty

pages will OCR at a varied accuracy rate before cleanup steps. There is a possibility that some dirty pages will require rekeying. These document characteristics present challenges for economical COTS software to cleanly convert paper to electronic format.

TO Document Characteristics:

- Black and white
- Many pages per volume
- Some division by chapters
- Double-sided pages
- Varied paper sizes
- Varied paper quality
- Majority machine typed characters
- Varied print quality — multi-generation copies
- Varied Format — font style and size, spacing, graphics, tables
- Handling — handwritten notations, highlighted, stapled, curled, and torn.

6.0 BASELINE CASE

6.1 CONVERSION PROCESS FROM MS WORD TO SGML

TAMSCO had researched several commercially available products to assist in the conversion from an MS Word file to SGML. This research began in the summer of 1995, and at that time, the most useful product at a reasonable price was the SGML Author for Word 6.0. During this research, there were only approximately eight tools that would convert from non-SGML to SGML. Of these tools, all were available for the Unix platform, however, only four were available through a Windows environment. The tool had to be usable through Windows since the government clients at Warner-Robins Air Logistics Center (WR-ALC) had requested that camera ready TOs be delivered not only as paper, but also in MS Word format. The available tools consisted of SGML Author for Word 6.0, OmniMark, Avalanche tools and products, and DynaTag. The cost limitation was $1000, and of these tools the only one that was under $1000 was the SGML Author for Word 6.0. Many of the available tools required several one week training sessions. SGML Author for Word was self explanatory enough that no training was required. Also, at that time, not all the tools could handle the complexities found in the USAF DTDs. Even today a few tools do not implement all the SGML functions found in the AF DTDs. While this product was affordable, it was ineffective. To perform a conversion, the native Word file had to be developed to an MS Word template. The MS Word template defined the text, graphics, and formatting (such as font size and style) in documents. It could, however, be easily modified by any user. In MS Word, it was up to the template to enforce the SGML rules defined in the DTD, but an MS Word template does not have this capability due to the fact that it was easily modified. Portions of the conversion were very effective. However, to make the SGML tagged instance a quality deliverable, human intervention was necessary. An analyst had to go into the SGML instance and correct any errors the conversion routine introduced. Each time an analyst edited the SGML instance, it was possible to introduce even more errors. An example of one of the errors this product introduced was to not place the required elements with the parent element such as leaving a required <title> off of a <para0>. To be compliant with the structure of the DTD, this type of error was not allowed because the SGML file would not parse if left this way.

For such errors, additional conversion processes such as multiple MS Word macros had to be developed. The MS Word macros were used to correct some of the minor tagging errors (for example a <para0> needed a <title>, and remove or add numbering to the attributes of the elements). The SGML analysts also had to perform a great deal of clean up to the converted files to achieve an accurate, Continuous Acquisition and Life Cycle Support (CALS) compliant SGML tagged instance that would parse. Over the past few years, this product has become more robust. However, as a stand-alone product, it is still not efficient enough.

While attending the CALS Expo '97 conference, the COPIS team found one organization that was having success using the SGML Author for Word 6.0. As part of an Army contract, a contractor developed many Visual Basic programs and Word Macros to make the SGML Author for Word 6.0 meet the requirements. The contractor said without their programs and macros the software would be useless. The macros and Visual Basic Programs had the ability to enforce the

SGML rules in the DTD, so the author could never input data that would not parse. Also instead of having an MS Word template for an entire manual, the templates were built at the element level or the information unit level. An information unit would be a definition list the term or the definition alone is meaningless, but when grouped together it provides useful information. The additional programming would use the correct template based on the SGML DTD.

A brief description of some of the software identified during the research is shown in Table 6-1.

Table 6-1. SGML Products

Product	Description
Context-Wise, by U.S. Lynx	It can transform ordinary word-processing documents, with all their embedded formatting commands, into SGML tagged documents. A table-driven tagger. It allows quick response to changing document layout and to differing DTDs. It has document-interpretation tools designed to handle hierarchical, nested text structures like SGML. Translation features: •Entire tables may be swapped while processing. •Up to six tables can be run sequentially in one batch process. Modular tables can be combined for different DTDs or a previous table can make a discrimination that sets up a translation alternate in a subsequent table. •Programmable character sets can be mapped directly to ISO character entity references. •Files can be merged and split up during the document translation. Files can be renamed automatically during translation to conform to specification requirements such as the CALS text file identifiers required by the 1840A specification.
EasyTag, by TetraSys	Made by a French company – only able to find French literature.
OmniMark, OmniMark Technologies Corp.	•SGML aware programming language. •Perform large-scale conversion and mark-up of text and data for delivery in print or on CD-ROM. •Allow Intranet and website builders to automate the dynamic presentation of individualized content.

Table 6-1. SGML Products (Continued)

Product	Description
Roustabout, by Apropos Toy & Tool Development	Quark-to-SGML translation utility, reimplemented in Java, was being beta-tested. Roustabout is a stand-alone application that translates QuarkXpress-formatted text into valid SGML. By default, it: • Creates SGML elements guided by text formatting and making use of Quark stylesheet names and definitions where possible. • Replaces each non-ASCII character in the original text with a general entity comprised of the current font name and the character position. • Produces a simple two-level structure, block elements containing inline elements, maintaining original paragraph breaks. The interface is straightforward and user-named option sets are maintained. Mapping files enable customization, including: • Specification of target element generic identifiers and attributes. • Substitution of general entities for characters of expert and dingbat fonts. Written in Java, Roustabout will run on most systems offering a Java Virtual Machine, including Macintosh, Windows95 and many versions of UNIX. It will convert files produced by QuarkXpress for Macintosh and Windows.

As an alternative to having small to medium sized companies perform a conversion from paper to SGML, the COPIS team began searching for conversion vendors. After searching the Internet, speaking with SGML consultants, and SGML implementers, several SGML conversion vendors were recommended. Searches on the Internet found Input Center, Active Systems, Delta Computers Ltd, Sencor, and Data Conversion Laboratories.

There were several companies found located around the world that convert paper to SGML. (Again, as more companies become proficient in the use of SGML, more companies will be able to provide the service of converting paper to SGML). Two companies provided information for this study.

A quote was received from Data Conversion Laboratory for 500 pages. There was no breakdown of processes that they utilize. The total cost would be $12,500 resulting from $10 per page conversion costs, plus a startup fee of $7,500. Data Conversion Laboratory did say the price would drop to $4 per page if the volume increased, but no quote as to how much more the volume needed to be.

For a point of reference, ActiveSystems provided a quote to convert 1,000 to 1,500 pages to SGML. There is an additional step of converting the paper to ASCII before the conversion to SGML. Depending on the rate of accuracy needed, the cost ranged from $5 to $10 a page. They

also provided a quote to convert paper to PDF, and the cost range on that was $0.75 to $6.55 per page. They do offer a discount for volumes of 100,000 or more pages. Details of the quote are outlined in the following paragraphs.

To convert from paper to ASCII with a character accuracy rate of 99.995%, the process involved a double pass of the documentation by two separate conversion teams. The steps in this process would be as follows:

- Paper material is prepared for scanning.
- Using an HP ScanJet 4C flatbed scanner and a Xerox K6200 flatbed scanner, the hardcopy material is scanned and ran through an OCR process to ASCII format by two separate teams.
- The two versions of ASCII files are transferred to a PC work station for the first edit and cleanup step (including spell-checking verification).
- The two versions of ASCII files are then electronically compared and an error list is generated. A single version of the files is selected for correction of all identified errors.
- The corrected ASCII files are then printed and visually compared by staff with the original documents. This 'Stare and Compare' step catches virtually all errors potentially introduced by the conversion process.
- Changes are noted on the printed documents and the ASCII electronic documents are corrected.

The steps for character accuracy of 99.95% which involves a single pass of the documentation instead of a double pass are:

- Paper material is prepared for scanning.
- Using an HP ScanJet 4C flatbed scanner or a Xerox K6200 flatbed scanner, the hardcopy material is scanned. Text is run through an OCR process into ASCII.
- ASCII files are transferred to a PC workstation for spell checking. The cleanup includes hyphenation correction (caused by lines of text which wrap from one line to the next), and OCR error correction.
- Text is then reviewed to ensure accuracy to 99.95%.

To obtain 99.5% character accuracy, only one pass of the documentation is required, and the text does not go through final review.

- Paper material is prepared for scanning.

- Using an HP ScanJet 4C flatbed scanner or a Xerox K6200 flatbed scanner, the hardcopy material is scanned. Text is run through an OCR process into ASCII.

- ASCII files are transferred to a PC workstation for spell checking. The cleanup includes hyphenation correction (caused by lines of text which wrap from one line to the next), and OCR error correction.

To convert from an ASCII file to an SGML file, the data would go through a review and analysis phase. Initially, a detailed analysis of the format and structure of a representative selection of documents scheduled for conversion would be conducted. The military standard MIL-STD-38784 DTD must also be analyzed.

A conversion specification, that would translate the ASCII components to the military standard MIL-STD-38784 DTD element structures, would then be prepared.

Documents, in ASCII markup, would then be mapped to the element structures of the military standard MIL-STD-38784 DTD, and special characters would be converted to SGML entities.

To convert from paper to PDF at a character accuracy rate of 99.995%, the process involves a double pass of the documentation by two separate conversion teams. The steps in this process are:

- Paper material is prepared for scanning.

- Using an HP ScanJet 4C flatbed scanner and a Xerox K6200 flatbed scanner, the hardcopy material is scanned to Adobe Capture Document format by two separate teams.

- The two versions of Adobe Capture Document files are transferred to a PC work station for the first edit and cleanup step (including spell-checking verification).

- A hardcopy printout of the corrected Adobe Capture Document file is then compared to the original. All errors are corrected in the Adobe Capture Document file.

- Adobe Capture Document files are then converted to PDF. If this material is intended for Web distribution, the PDF files are then optimized. For large books, files may be grouped for processing and combined for final delivery.

For a character accuracy rate of 99.95% the process involves a single pass of the documentation. The steps in this process are:

- Paper material is prepared for scanning.

- Using an HP ScanJet 4C flatbed scanner or a Xerox K6200 flatbed scanner, the hardcopy is scanned to TIFF format.

- Tagged Image File Format (TIFF) files are converted to Adobe Capture Document format for cleanup. Editors review the material and perform a spell check.

- Adobe Capture Document files are converted to PDF. If this material is intended for Web distribution, the PDF files are then optimized. For large books, files may be grouped for processing and combined for final delivery.

For a character accuracy rate of 99.5% the process involves a single pass of the documentation. The steps in this process are:

- Paper material is prepared for scanning.

- Using an HP ScanJet 4C flatbed scanner or a Xerox K6200 flatbed scanner, the hardcopy material is scanned to TIFF. Graphic images are inspected for clarity and re-scanned if required.

- TIFF files are converted to PDF. If the material is intended for Web distribution, the PDF files are then optimized. For large books files may be grouped for processing and combined for final delivery.

The documents will be indexed and bookmarked with Adobe Acrobat Exchange. Each file will be bookmarked with a Table Of Contents (TOC) that corresponds to that of the hardcopy document. Depending on document organization, chapter, section, or paragraph bookmarks could also be included.

A summary of the pricing based on the type of conversion and to what character accuracy level is in Table 6-2.

Table 6-2. Conversion Service Bureau Pricing

Hardcopy Conversion to ASCII / SGML	Per Page
Hardcopy Conversion to ASCII at 99.995% and SGML	$ 10.00
Hardcopy Conversion to ASCII at 99.95% and SGML	
Hardcopy Conversion to ASCII at 99.5% and SGML	$ 5.00
Hardcopy Conversion to PDF	Per Page
Hardcopy Conversion to PDF at 99.995%	$ 6.55
Hardcopy Conversion to PDF at 99.95%	$ 3.75
Hardcopy Conversion to PDF at 99.5%	$ 0.75

In addition to converting from paper to SGML, TAMSCO has found the following information based on converting 1,224 pages from a digital format to SGML tagged data, including both text and graphics, as well as an additional 532 blank pages for the change packages that were part of a delivery. (A total of 1,756 pages were converted from digital files to SGML tagged instances.) Those costs are outlined in Table 6-3.

Table 6-3. Paper to SGML Costs

Labor Category	Rate	Hrs	Total	Pages	Cost/Page
PM	84.44	27	$2,279.88		
PMTS	62.71	88	$5,518.48		
SMTS	52.04	92	$4,787.68		
DET	24.55	40	$982.00		
Total		247.00	$13,568.04	1,224	$11.09

The hardware used for this effort included two Pentium computers with 1.7 GB hard drive space, 16 MB of RAM, and a 486 PC. The software is described in Table 6-4.

Table 6-4. Software Used in SGML Conversion

Product	Description
Word 6.0	The digital data was authored in Word 6.0. Templates had previously been developed. Three templates were used in this conversion.
Sgmls	shareware parser
Exoterica Validator	This parser is part of the OmniMark tool suite.
JCALS System	The AF PDSM program office composed several of the TAMSCO developed SGML Tagged instances.

The parsers, described in Table 6-4, check for completeness of the document in SGML terms. The parser checks to ensure the document follows the rules of the DTD. For example, it ensures that every <para0> is followed by a <title>, if that is what the DTD requires. It does not, however, check for completeness in terms of the text of the document.

6.2 CONVERSION PROCESS FROM SGML TO IETM

The following are the classification levels of IETMs published by the Caderock Division Naval Surface Warfare Center.

Group I consists of the basic class (page turners). Class 0 IETMs are electronically indexed page images for digitized viewing, but not navigation. They are intended for electronic archival filing or Print-on-Demand. These can be viewed on an electronic display, but have no detailed index for navigation through the document for purpose of on-line usage. Class 1 IETMs are electronically indexed page images, which are digitized, allowing user access. They are different from class 0 IETMs in that they are indexed for navigation.

Group II consists of the advanced class (Scrolling Hypertext). Class 2 IETMs are electronically scrolling documents for interactive display of ASCII documents. They use an intelligent index and Hypertext tags inserted into a tagged document file. The class 3 IETMs are linear structured IETMs for interactive display of data, which is tagged using SGML tags.

Group III consists of the extended classes (Interactive Database). Class 4 IETMs are hierarchically structured IETMs, which allow for interactive display of data authored into a non-redundant relational database. Class 5 IETMs are an integrated database which integrate electronic technical information systems for display of class 4 IETMs with other type technical data and processes like that of an expert system.

TAMSCO had previously converted SGML tagged TOs to a proprietary IETM format. The IETM format is Hughes Technical Services Company's (HTSC), now Raytheon, Advanced Integrated Maintenance Support System (AIMSS). AIMSS is a windows based software package that has an authoring toolkit and runtime software. The DTD used by the AIMSS product is a modified version of the existing MIL-D-87269. Based on the classification levels published by the Caderock Division Naval Surface Warfare Center, this product is a class four IETM and was utilized due to a mentor/protégé relationship between Hughes Aircraft Company and TAMSCO. Since other IETM products, including the current DOD IETMs such as Metafile for Interactive Documents (MID), also modify the MIL-D-87269 DTD, it is assumed that the findings would be similar for the other systems.

The hardware used during this conversion process consisted of a Pentium 166 MHz processor with 32 MB of RAM with a 2.0 GB hard drive. The software included the AIMSS product and an ASCII text editor. The goal was for the IETM to look like the paper manuals in style and format, but to operate as an IETM. Therefore, the paper copies of the TOs were also available to ensure the IETM was consistent with the paper. There were approximately 15 TOs considered for the SGML to IETM project. The data was analyzed into systems and studied to determine if and how the manuals referenced one another, and the number of TOs diminished. The data consisted of general equipment manuals, general system manuals, and job guides. A storyboard was developed based on the remaining TOs to organize the conversion effort. From this storyboard, duplicate data was eliminated, and the remaining data was ported into the AIMSS product.

After the data was in the AIMSS format, words such as "manual" and "chapter" were modified to reflect a more electronic form as opposed to a paper print out. The context of the data was not changed. Manual hyperlinks had to be created to traverse through the IETM. There were three basic types of links used: goto, holdup, and stayup.

There are two types of goto links, one takes the user to a descriptive topic, and the other takes the user to what is called a task object. An example of a descriptive topic is a safety summary or general information, and a task topic is something of a procedural nature like removal and installation.

A holdup link brings the data up in an additional window, which appears in front of the current screen. Once the user clicks onto another screen, the window that was linked to is dismissed.

A stayup link also brings up an additional window in front of the current one, but it has to be dismissed by the user. This is different from the previous link in that this one has to be closed

out of like any windows based program. The holdup link is automatically closed when the user selects something else.

All of the links in the IETM were manually created. The index and the TOC also had to be manually created. Then the file had to pass a quality assurance review and an aircraft analyst review for usability and accuracy.

The AIMSS product will import an SGML document that is tagged in accordance with the DTD used with AIMSS. The SGML files being used in this case were not tagged to this particular DTD. Instead, the files were tagged to the AF specification MIL-PRF-83495B. There is no automatic or "hands off" method to convert from MIL-PRF-83495B DTDs to the AIMSS DTD. However, it was found that Omnimark (an SGML aware programming language) could be used to write a conversion routine. This conversion routine would ultimately translate the MIL-PRF-83495B DTD to the DTD the AIMSS product utilizes. It would still be necessary to do some minor cleanup in terms of minor linking issues and traversing the IETM. The OmniMark conversion routine was not created for this effort.

This project team consisted of a Primary IETM Analyst (PMTS), a Senior IETM Analyst (SMTS), a Senior Illustrator (SMTS), a Technical Writer/Analyst (MTS), and initially a Consultant. Table 6-5 illustrates the cost per page of this particular conversion effort with the consultant and without.

Table 6-5. Cost of IETMs

Labor Category		Hrs	Rate*	Total	Pages	Cost/Page
SMTS		177.00	52.04	$9,211.08		
MTS		10.00	38.24	$382.40		
PMTS		60.00	62.71	$3,762.60		
Consultant		109.00	62.71	$6,835.39		
SGML Tag ($11.09/pg)				$1,928.79		
Total		356.00		$22,120.26	174	$127.13
SMTS		200.00	52.04	$10,408.00		
MTS		15.00	38.24	$573.60		
PMTS		28.00	62.71	$1,755.88		
Consultant		0.00		$0.00		
SGML Tag ($11.09/pg)				$2,050.73		
Total		243.00		$14,788.21	185	$79.94

** Rates are based on the TAMSCO GSA Schedule, Zone 3*

After researching other sources of class four IETM conversion, it was found that the cost per page figures were in line with the industry averages. A verbal quote from HTSC indicated they would charge $100 to $150.00 per page. "Interactive Electronic Technical Manual (IETM)

Process Plan; Processes and Guidance for IETM Implementation" published by Commander, Naval Sea Systems Command and Commander, Space and Naval Warfare Systems Command on 5 December 1995 estimated costs to be near $100 per page, but down to around $40 per page by the year 2000.

A description of the products associated with the IETMs quotes discussed earlier, is in Table 6-6.

Table 6-6. Products Used in IETMs Pricing

Product	Description
Advanced Integrated Maintenance Support System, by Hughes Technical Services Company	• Windows based environment. Runs under Windows 3.X or Windows 95 • Fully integrated authoring and run-time software. Allows authors to instantly see how the IETM will appear. • CGM, WMF, and BMP graphic capabilities. • Compliance with MIL-M-87268 and modified MIL-D-87269 • Authoring and Runtime system
OmniMark, OmniMark Technologies Corp.	• SGML aware programming language • Performs large-scale conversion and mark-up of text and data for delivery in print or on CD-ROM • Allows Intranet and website builders to automate the dynamic presentation of individualized content

In determining if there were other commercially available class four IETM tools available that would make converting to this format easier or less expensive, it was discovered that there is at least one prototype being developed by Aquidneck Management Associates Ltd. in conjunction with the Tri-Service IETM Working Group. The product itself is due to be completed within the next twelve months. This project is an attempt to bring the different implementations of the IETM specifications together to produce a more standardized IETM format and provide enhancements to the next version of the military specification MIL-PRF-87269.

7.0 TECHNOLOGY ASSESSMENT

7.1 PAPER FORMAT CONVERSION TO ELECTRONIC FORMAT

As described in the Methods, Assumptions, and Procedures section of this document, the following conversion methods have been evaluated:

- Paper to IETMS (MS Word to SGML and SGML to IETMS)
- Paper to HTML (Paper scanned and converted by OCR to MS Word and MS Word to HTML)
- MS Word (authored) to HTML
- SGML to HTML
- Paper to PDF.

COTS software package searches via Computer Select and the Internet were based on the criteria of pricing less than $2,000 and several search term combinations such as, "HTML", "SGML", "PDF", "IETMS", "conversion", "linking", "OCR", "versioning", "configuration management", "natural language", and names of software companies (identified in periodicals and COPIS team members' knowledge of the industry). COPIS team members performed weekly searches as the technologies being researched are evolving quickly with new and improved software product introductions and upgrades.

A secondary review of software package search results narrowed down software by the following criteria:

- Platform = Windows NT, Win95
- Package Features/Descriptive Function
- Company Stability
- Year Product Developed
- Number of Packages Sold
- Third Party Evaluations & Reports — Bruce Silver & Associates, Seybold, Doculabs, Editor's Choice.

Many of the software package descriptions claimed to convert paper and various digital formats into HTML files. However, most of these packages were low-end page viewers and HTML page authoring/editing tools with no additional capabilities such as hyperlink insertion or pattern matching. Only eight software packages remained after this evaluation. Additionally, two SGML to HTML conversion software packages were evaluated. A total of ten packages in

Table 7-1 were evaluated for usability of features and quality of their electronic output. The costs of the various packages ranged from $39.00 to $2,000.00, except ArborText, which is one of the few packages available to convert SGML to HTML. ArborText sells a license for under $2,000, but requires a UNIX server and server software for an additional cost (total of $5,000).

Based on the results, evaluation of six of the ten software packages continued by testing packages with the source documents and grading the packages using a point system of weighted criteria and numeric values. The best software packages were used to design a conversion process, a demonstration of a technical maintenance scenario in HTML accessed via Netscape or MS Internet Explorer, and a configuration management plan.

Table 7-1. COTS Packages for Preliminary Evaluation

Software Package	Upgraded during COPIS	Purpose	Cost	Procurement
Adobe Acrobat Capture 1.0	Upgraded to 2.0	Paper to DOC, PDF, and HTML	$895	TASC owned
Adobe FrameMaker + SGML		SGML to HTML	$2,000	Evaluation copy
ANT		HTML	$39	Trail Download
ArborText		SGML to HTML	$5,000	TAMSCO owned
Easy Help/Web		HMTL	$280	Trail Download
Info Access HTML Transit 2.0	Upgraded to 3.0	HTML	$495	COPIS Purchase
Caere OmniPage 7.0	Upgraded to 8.0	Paper to DOC and HTML	$499	TASC owned
Xerox Pagis Pro (Textbridge 96)	Upgraded to TextBridge 98	Paper to DOC Paper to DOC and HTML	$99 $82	Trail Download COPIS Purchase
Web Publisher		HTML	$495	Trail Download
Caere WordScan Plus 4.0		Paper to DOC	$595	TASC owned

Three categories separated the functionality of the software packages: Scanning and OCR, HTML Translation and Linking, and SGML to HTML.

7.1.1 Scanning and OCR

Four software packages were evaluated for their capabilities to convert paper to computer readable format. These packages, listed in Table 7-1, are Adobe Acrobat Capture, Caere OmniPage, Xerox Pagis Pro (TextBridge), and Caere WordScan Plus. OCR software packages work in three basic steps. First, each paper page is converted to an image format (capture process). Second, the image format is analyzed by a series of algorithms to determine individual

characters on the page (OCR process). Third, those characters are converted to a usable electronic file type such as ASCII, popular word processing formats, or HTML (conversion process).

The HTML conversion capabilities have been introduced in some of the COTS OCR software packages during the timeframe of COPIS. Evaluation of the HTML output from the paper source documents has been included in the OCR and Scanning.

For the purpose of evaluating the quality of OCR and conversion steps of the four OCR packages, a single capture process has been defined.

The capture process is a matter of gathering the paper information and converting it to a format that OCR software algorithms can understand. Scanning creates a digital image (picture) of paper. A simple analogy to scanning would be using a copier to duplicate the original, but instead of paper output from the copier, the scanned output is an electronic file routed to a predetermined location on a computer.

OCR is a process by which the image file of machine print information is converted into a text file. For example, the letter "c" in an image is a cluster of black dots that form the shape "c". OCR looks at the dots to determine what the groups of dots represent as characters typed from a keyboard. Based on complicated algorithms and some artificial intelligence, OCR software can convert images to text at high speeds; much faster than a data entry person can retype a document or determine identification fields for a document. However, OCR accuracy is very dependent on the quality of paper originals.

The scan density determines image quality. Scanning is typically conducted at 300 Dots Per Inch (DPI) in both directions. Higher scan densities are needed when the scanned pages contain fine detail such as very small print and fine lines. This is necessary so that the size of the scan dot can easily resolve black space from white and result in a clear definition of the information. Scan density for all tests was set to 400 DPI.

The paper scanner used in testing was the Fujitsu M3099 production level scanner with an Image Processor Circuit II combined with a Kofax KF-9275 High Speed Image Processor Board. The scanner is attached to a Pentium/P5-90 PC through a Kofax KF-9275 advanced document processor board and RS232C/video cable. Typically, the KF-9275 can reduce a 1MB image to 50KB or less. This image file supports the Consultative Committee on International Telegraphy and Telephony (CCITT) Group 4 file format with a standard TIFF header format.*

---

* *The Consultative Committee on International Telegraphy and Telephony (CCITT) was established to recommend worldwide communications usage standards. There are currently two international standards available (CCITT Group 3 and 4). The COPIS capture process will store scanned images as a Tagged Image File Format (TIFF). Use of non-proprietary file headers is necessary to import the same image files into each COTS OCR package being evaluated. TIFF images are based on the CCITT standard and even though TIFF is not necessarily recommended as a standard, TIFF is considered a defacto standard for header label conventions.*

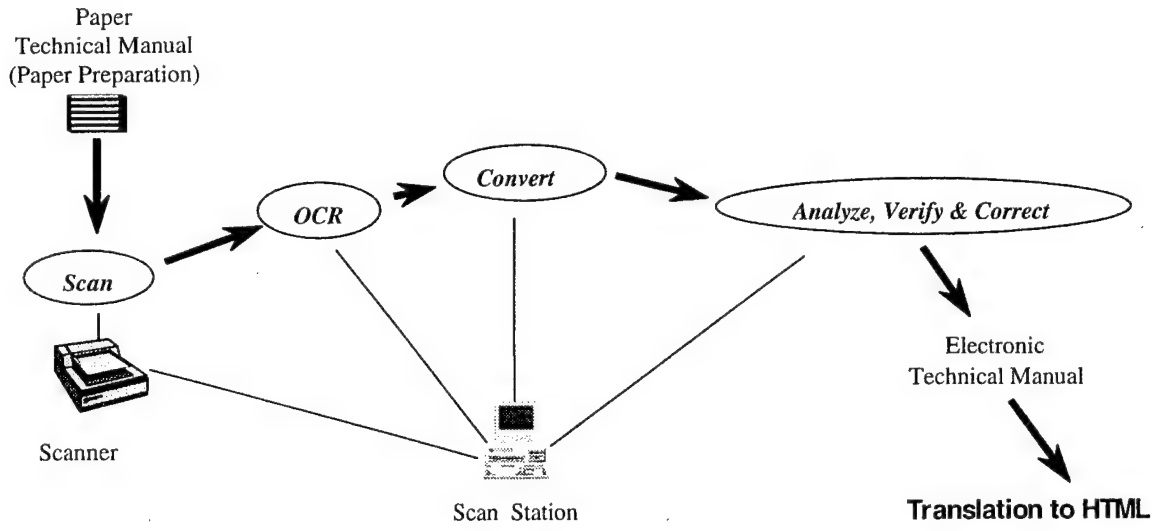


Figure 7-1. Scan and OCR Process

The Fujitsu M3099 scanner is rated at 55 pages per minute (55 images single-sided or 110 images double-sided) for documents sized 2.5" x 3" up to 11" x 17". It has a built-in 500-sheet automatic document feeder to handle paperweights ranging from 14 to 34 pounds. Since user selectable resolutions of 200, 240, 300, and 400 DPI are stored by document type, applications and documents can be tuned for optimum document throughput and image quality. After scanning, the document image can be displayed in portrait format at the scan workstation monitor within one second.

Using this hardware and process, all paper pages were scanned once and source documents were imported as TIFF images into Adobe Acrobat Capture 2.0, Caere OmniPage 8.0, Caere WordScan Plus 4.0, and Xerox ScanSoft TextBridge 98.

7.1.2 Preliminary Evaluation of OCR Software Packages

The preliminary evaluation of the OCR software packages centered on two factors, conversion formats and the output quality of OCR processed and converted files. Table 7-2 depicts which software packages had capabilities to output to the file formats required for COPIS.

Table 7-2. OCR Package Electronic Output Formats

	Adobe Acrobat Capture 2.0	Caere OmniPage 7.0	Caere WordScan Plus 4.0	Xerox Pagis Pro (TextBridge 96)
Output Types (Text)				
DOC	X	X	X	
RTF	X	X	X	X
HTML	X	X		X
PDF	X			
Output Types (Graphics)				
TIFF	X	X	X	X
JPEG	X			X
GIF		X		

For each output format, available test images from the source documents were OCR processed and converted. The evaluation of the usability of the package was based on the resulting output. The assessment of an output file was based on the number of resulting character errors, style formatting, and graphics retention. If the appearance of the output obviously required more effort to edit than it would to rekey the information, the file was labeled unusable.

Following are the results of the preliminary OCR process and conversions. Dirty paper is defined as degenerated copies or fax copies of information containing ink smears, filled and darkened characters, or lightened and broken characters. Clean paper is defined as original laser printed quality with no extraneous markings. Additionally, the Adobe PDF file format was evaluated for conversion to alternate file formats. More information regarding the evaluation of PDF follows these results in Section 7.1.3.

Results of the preliminary evaluation of Adobe, Acrobat Capture 2.0, Xerox, Pagis Pro (TextBridge 96), Caere, OmniPage 7.0, and Caere, WordScan Plus 4.0 are detailed in Table 7-3.

Table 7-3. Preliminary OCR Conversion Results by File Type

Adobe, Acrobat Capture 2.0		
Input	**Output**	**Result**
Dirty Paper	PDF (Image)	Acceptable for viewing. No OCR performed for conversion. Comparable to a TIFF image.
	PDF (Normal)	Acceptable for viewing, no control of bitmap portions within and as a substitute for text for possible automatic hyperlinking.
	RTF or DOC	Unusable without significant cleanup.
	HTML	Unusable without significant cleanup.
Clean Paper	PDF (Image)	Acceptable for viewing. No OCR performed for conversion. Comparable to a TIFF image.
	PDF (Normal)	Acceptable for viewing, no control of bitmap portions within and as a substitute for text for possible automatic hyperlinking.
	RTF or DOC	Usable with minor cleanup for import to HTML conversion software.
	HTML	Usable with minor cleanup for an HTML editor and hand inserted links.
PDF (Image) - Clean	PDF (Normal)	Acceptable for viewing, no control of bitmap portions within and as a substitute for text for possible automatic hyperlinking.
	RTF or DOC	Usable with minor cleanup for import to HTML conversion software.
	HTML	Usable with minor cleanup for an HTML editor and hand inserted links.
PDF (Normal) - Clean	RTF or DOC	Software incapable of converting PDF (Normal) input. Must use Adobe proprietary ACD format.
	HTML	Software incapable of converting PDF (Normal) input. Must use Adobe proprietary ACD format.

Table 7-3. Preliminary OCR Conversion Results by File Type (Continued)

Xerox, Pagis Pro (TextBridge 96)		
Input	**Output**	**Result**
Dirty Paper	RTF	Usable with minor cleanup for import to HTML conversion software.
Clean Paper	RTF	Usable with minor cleanup for import to HTML conversion software.
Caere, OmniPage 7.0		
Input	**Output**	**Result**
Dirty Paper	RTF or DOC and HTML	Usable with minor cleanup for import to HTML conversion software.
Clean Paper	RTF or DOC and HTML	Usable with minor cleanup for import to HTML conversion software.
Caere WordScan Plus 4.0		
Input	**Output**	**Result**
Dirty Paper	RTF or DOC	Usable with minor cleanup for import to HTML conversion software.
Clean Paper	RTF or DOC	Usable with minor cleanup for import to HTML conversion software.

7.1.3 PDF Format Evaluation

HTML 3.2 was the specified file format of COPIS for viewing technical information through a commercial web-based browser. Adobe's PDF was identified as a potential alternative to HTML and is capable of being viewed through one of the COPIS suggested browsers utilizing Adobe Acrobat Reader 3.0. Acrobat Reader software can be downloaded as a plug-in from the WWW free of charge. This plug-in allows users to view, navigate and print PDF files. Acrobat software also provides an open architecture to integrate with a wide variety of applications.

There are several DOD government agencies converting paper information to PDF and the question to use PDF versus HTML is an ongoing issue in the electronic publishing industry.

PDF is a cross-platform file format that preserves document fidelity across all major computer platforms and printers. Through COTS viewing software, users of standard Windows®, Macintosh®, UNIX® and DOS computer systems can easily access PDF files.

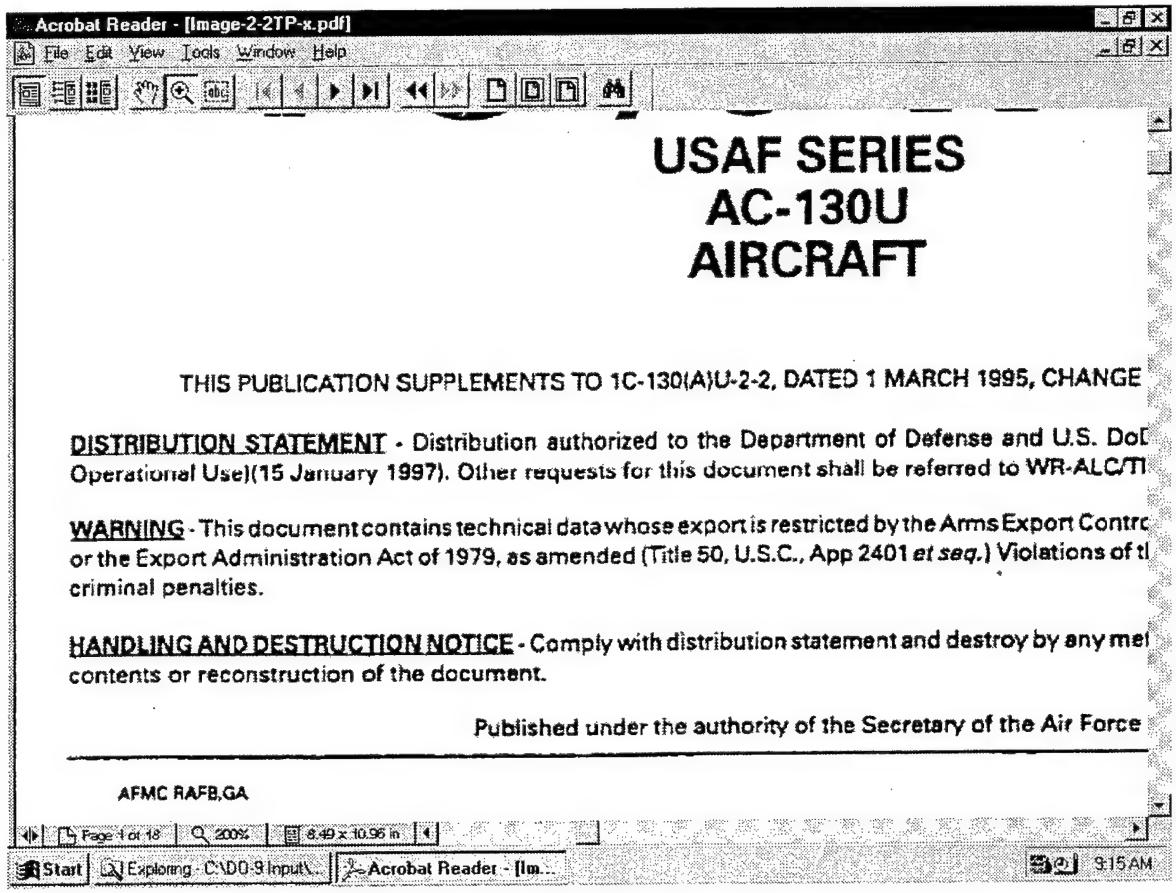


Figure 7-2. PDF (Image) Example

There are two versions or flavors of the PDF extension: PDF (Image), displayed in Figure 7-2 and PDF (Normal), displayed in Figure 7-3. An Adobe format related to PDF is a reviewer file with the file extension of ACD (Adobe Capture Document). PDF (Image) is sometimes considered an alternative to TIFF images. PDF (Normal) is a combination of text, bitmap representation of text and embedded graphics. PDF (Normal) enables systems' full-text search capabilities and usable document functions (such as cut and paste) possible for poor quality OCR originals. The PDF (Image) format is retrieved through an image viewer. Images can be viewed similar to turning pages of a book. No searching or linking capabilities were available. Documents looked exactly like the originals after they were converted, but were not exactly useful for anything but viewing in order from start to finish.

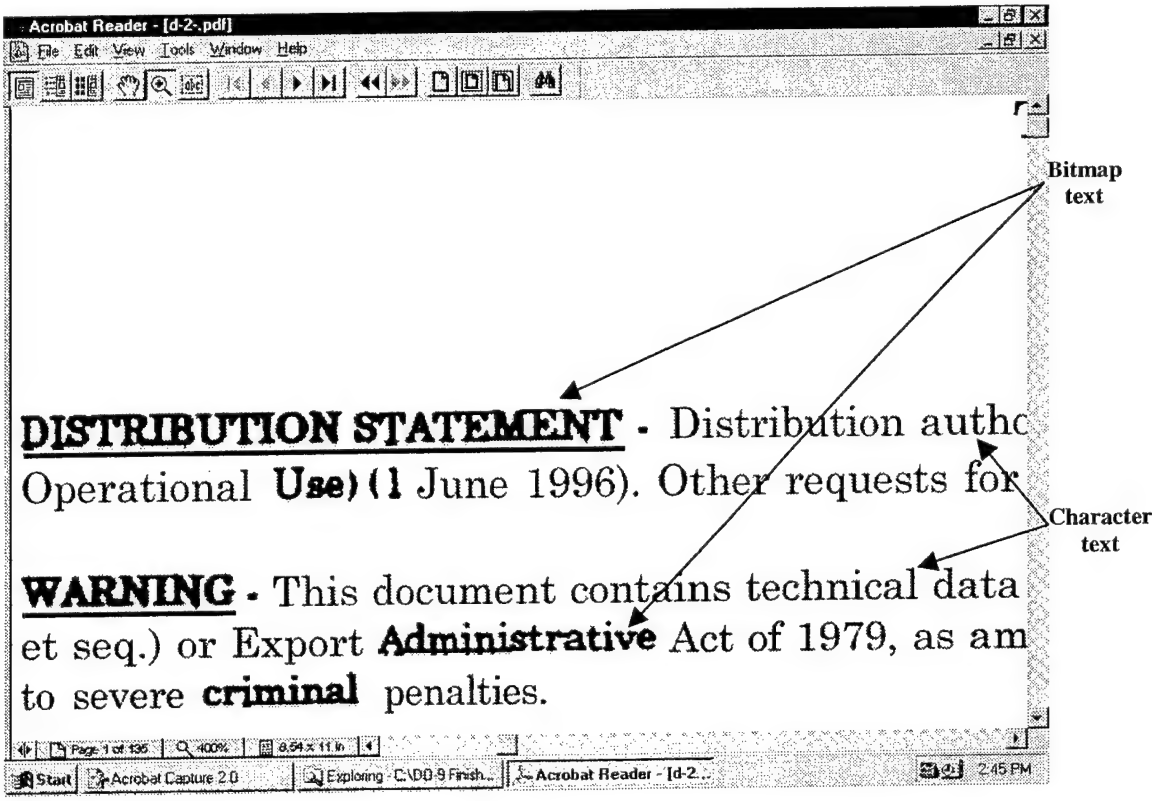


Figure 7-3. PDF (Normal) Example

Adobe Acrobat Capture 2.0 software uses technology to convert printed documents into accurate, searchable PDF (Normal) files that look exactly like the original printed page and are accessible on any platform. The text remains easily readable, reproduced in digital form without mistakes. Adobe Acrobat Capture is a document recognition software that, like OCR software, converts bitmapped page images acquired from a scanner into electronic text. But Adobe Acrobat Capture also preserves the character formatting, page layout, and graphics from the original paper pages. Two-column pages remain two-column pages; headlines remain headlines; and tables, charts, graphs, illustrations, and photos are preserved.

Adobe Acrobat Capture includes a module for reviewing the OCR processed document prior to conversion. The file extension for the OCR processed but unconverted file is ACD. This file is Adobe proprietary and is not usable by any software except the Reviewer. Figure 7-4 depicts an Adobe OCR processed portion of a TO in the Adobe Acrobat Reviewer format. The purpose of the reviewer is to make corrections to the characters, spelling, or formatting. Each color highlight indicates a potential error.

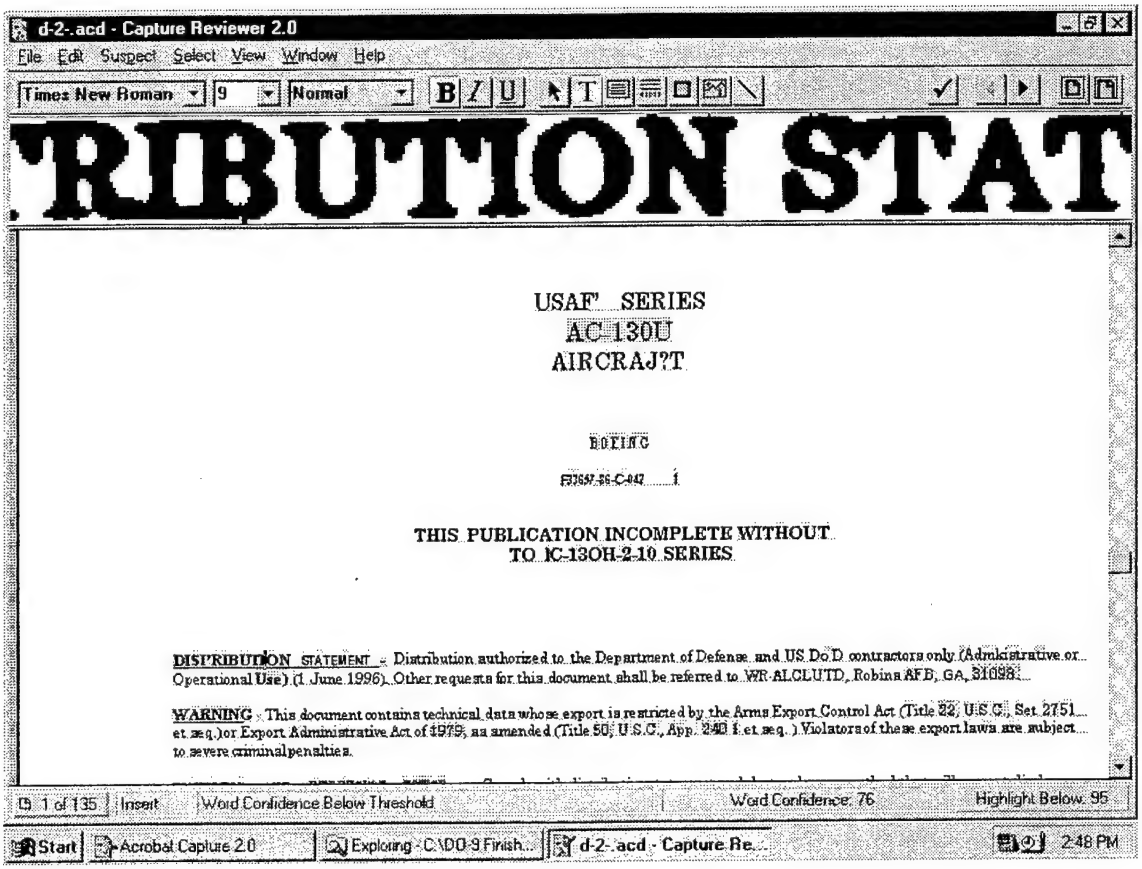


Figure 7-4. ACD Format Example

All four OCR software packages evaluated had some type of preconversion proofing or editing capability. There are at least two advantages to editing in this step. The first is to view the original image of the characters in order to make accurate corrections. The second is that corrections are made once before converting to multiple output formats. For example if a document is converted to an MS Word file and an HTML file, without the OCR proofing step each converted file would require individual proofing in its respective software package. In the case of degenerated originals and poor OCR output, the proofing and editing could exceed the costs of rekeying a document. Converting an OCR processed document into the PDF (Normal) format saves the cost of correcting questionable characters. Questionable characters are replaced by the original bitmap of those characters.

The ability to convert OCR processed documents can be delayed by saving the ACD file for each document. Therefore, a document scanned, OCR processed, and, proofed and saved in Reviewer can be converted to PDF (Normal), MS Word or another format at a later date.

The results of the PDF conversions can be found in the preliminary results of the OCR software evaluation in Table 7-3. Below are example outputs from Adobe Capture paper to PDF conversions without editing in the Adobe Reviewer. As expected, the PDF (Normal) file pictured in Figure 7-5 contains text and bitmaps of text. The representation of the information processed in Adobe Capture looks the same as the paper information. As a comparison, Figure 7-6 is the same information processed by Capture, but converted to an MS Word file. The Word conversion exhibits multiple errors in the character text as well as portions of text saved and inserted as an MS Word picture (graphic). Figure 7-7 displays a complex diagram common to TOs.

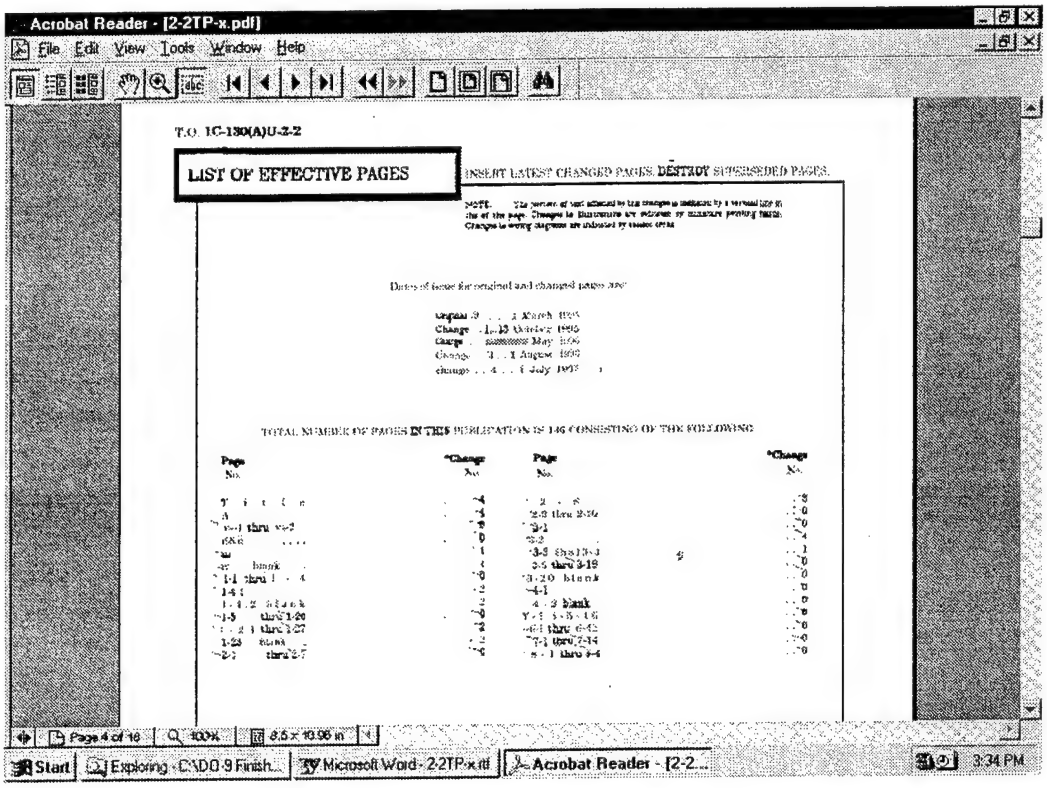


Figure 7-5. PDF (Normal) Conversion of a Table Example

Adobe Acrobat Capture does not provide a feature for the operator to designate which portions of a page are text or graphic. When processed in Adobe, the graphic contained in the Figure 7-7 was converted to partial text. Within electronic TOs, text associated with a graphic must be retained as part of the graphic. As with the table pictured above, a conversion to a format other than PDF (Normal) will produce chunks of graphics (text and drawing representations) and character strings randomly on a page. Figure 7-8 is an example of Capture's conversion to HTML. Using the reviewer (ACD format) would allow the correction of these pages. However, the effort required for editing this file will exceed the cost of manual page reconstruction.

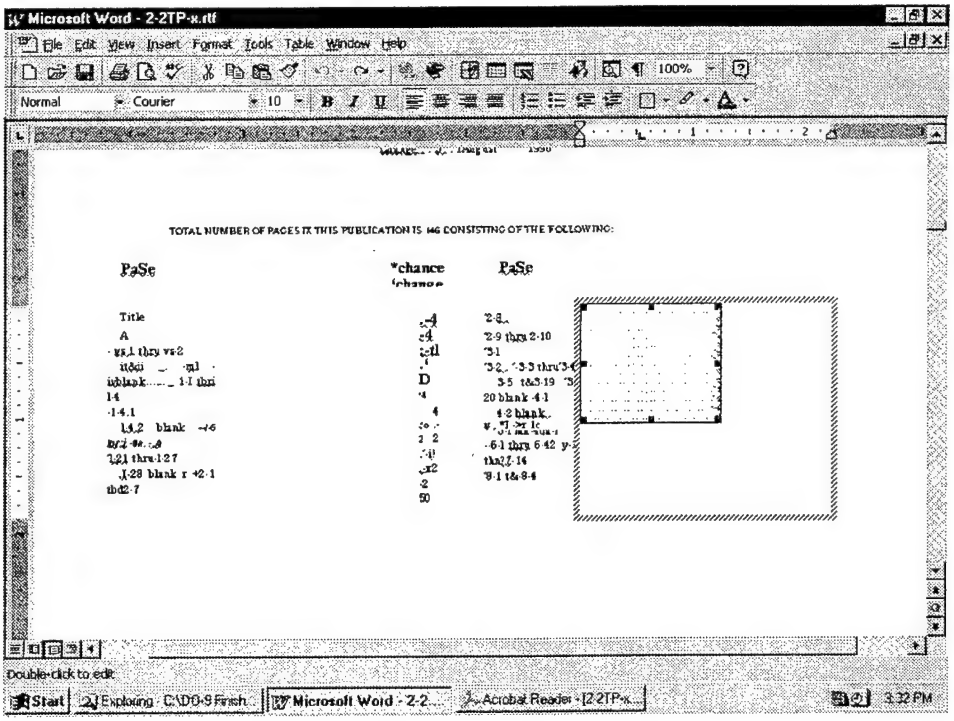


Figure 7-6. Adobe Capture Conversion to MS Word Example

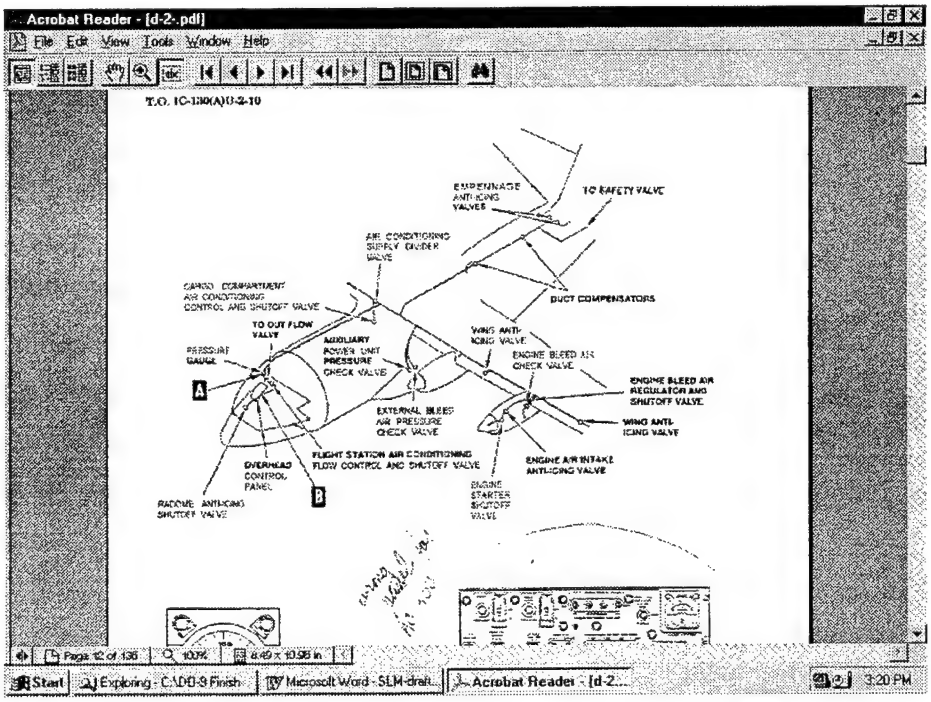


Figure 7-7. Complex TO Figure Converted to PDF (Normal) Example

7.1.3.1. PDF and Linking

As mentioned, Adobe Reader is available on the Internet at no charge for end users to view PDF files within a Web browser. The COPIS team assessed the features of Reader within Netscape. Bookmarks are links between sections marked by a heading in a PDF file to provide users with an automated table of contents. The COPIS team could not find the capability within Adobe Reader or Adobe Acrobat Capture to create bookmarks. Also, no hyperlink capabilities were found within Reader.

Another product, Adobe Exchange is expected to allow for insertion of bookmarks and hyperlinks. Using the Exchange product, each PDF (Image and Normal) can be manually bookmarked by headings or TOC, and PDF (Normal) file can be manually hyperlinked to information within a PDF (Normal) file or to another PDF (Normal) file.

It was found that in performing full-text searches within PDF (Normal) files, the search term was missed if it contained bitmap representations of text. Even though the PDF (Normal) file can be viewed with the exact likeness of the paper version, the usability of the file for search terms to create hyperlinks and find information is diminished due to the bitmaps.

The COPIS team found no other products capable of importing PDF files to generate hyperlinks, manually or automatically.

7.1.3.2. Converting PDF to HTML

An additional area researched by the COPIS team regarding PDF is the availability of PDF conversion software. At the time of COPIS, there is no COTS software available to convert PDF to other electronic formats. Alliances are currently being formed between Adobe and other vendors to incorporate the capability to imbed Adobe Viewer and Adobe Exchange software for use within their software (such as Adobe Reader 3.0 plug-in used in Netscape). PDF files cannot be viewed, edited, or saved in current word processing packages.

As described above, Adobe Acrobat Capture software has the capability to convert paper information to popular electronic output formats other than PDF (Normal). The COPIS team has not found any COTS software packages with the ability to import a PDF (Normal) file into non-Adobe products or plug-ins.

The only PDF conversion to HTML found was Adobe's Access conversion application which is available to comply with the Federal Disabilities Act.* The following Adobe Universal Resource Location (URL), http://www.adobe.com/prodindex/acrobat/advform.html, provides translation of PDF files to HTML for this purpose.

While at the Adobe site a user must enter the URL of the PDF document to be translated to HTML. A PDF file in a URL location that is accessible by the Adobe URL is the only way to

* PDF conversion information found in *Government Computer News* June 16, 1997 v16 n16 p48 (2) Challenged Web users have options to knock down access obstacles, Author, McCormick, John.

load the PDF file into the converter. Locations storing PDF files with security cannot be accessed by Adobe. For this scenario, users can download a version of Acrobat Reader from Adobe's web site that contains plug-ins to install the Adobe Access software locally.

The COPIS team was able to translate sample PDF files located at URLs behind Internet firewall server security. Translation of one and two page PDF documents took seconds. All graphics and formatting were lost. Since the graphics are imperative for use in TOs, this method of PDF translation to HTML was abandoned for further study.

7.1.3.3. PDF Recommendation

According to Bruce Silver and Associates, Industry Trend Reports, October 1997, Automated Web Publishing, Transit Central Streamlines Content Conversion and Management,

> "Desktop authoring tools are oriented toward printed output, fixed-size pages, and sequential reading, while the web model emphasizes random access of short hyperlinked "pages". HTML embodies the web model, usually more convenient than a replica of the printed document for interactive viewing. Alternative rendition formats attempt to provide better fidelity to the printed layout and pagination of the originally authored document.
>
> While HTML does lack these alternative formats' ability to precisely reproduce certain features of the printed output, it remains the predominant web-publishing format for several reasons:
>
> - Viewing is ubiquitous and free, native to all web browsers on all platforms, without extra plug-ins or downloaded applets.
> - It usually provides the smallest file size and the fastest first-page access.
> - It is open and supported by all third party tools, not controlled by a single vendor.
> - It is oriented to interactive screen viewing, the principal mode of web usage."

From an industry perspective, PDF is most appropriate for long documents that are intended to be saved or printed after downloading, or where reference to the original page numbering is important, or for documents where precise control of the layout by the author is required.

In the current TO environment, it is the content of the document, the technical procedures and the relationships to other technical procedures within a manual and between manuals that are important. The look and style are important for ease of use; however, hyperlinking between tasks and accessibility to the tasks while performing maintenance are the more important.

The COPIS team continued focus of the additional research on the conversion of paper to HTML for use of viewing technical manuals through a COTS web browser.

The AF PDSM effort mentioned in section 3.0 Methods, Assumptions, and Procedures, is currently converting paper technical information to a PDF format and to the interim Adobe Reviewer ACD format. Since PDF is proprietary, preserving the ACD format will allow for future cleanup and conversion of the PDF (Normal) files resulting from AF PDSM. As well, if conversion from PDF to other formats remains difficult, AF PDSM should have the ability to use the ACD files for conversion to formats other than PDF.

7.1.4 OCR Conversion to HTML

With the growing popularity of Web publishing, OCR and conversion software packages are including the option to convert to HTML. Three of the four OCR software packages evaluated during the time frame of COPIS had upgrades. Each upgrade included conversion to HTML. The COPIS team evaluated this potential in order to eliminate the need for a separate translation from a word processing format to HTML.

Figure 7-8 is an example of Adobe Capture's conversion to HTML. As in the conversions detailed in the PDF evaluation, Adobe's does not provide the feature to manually define text and graphics on a page. Again, using the reviewer (ACD format) would allow the correction of these pages. The effort required for editing will exceed the cost of manual page reconstruction in an HTML editor.

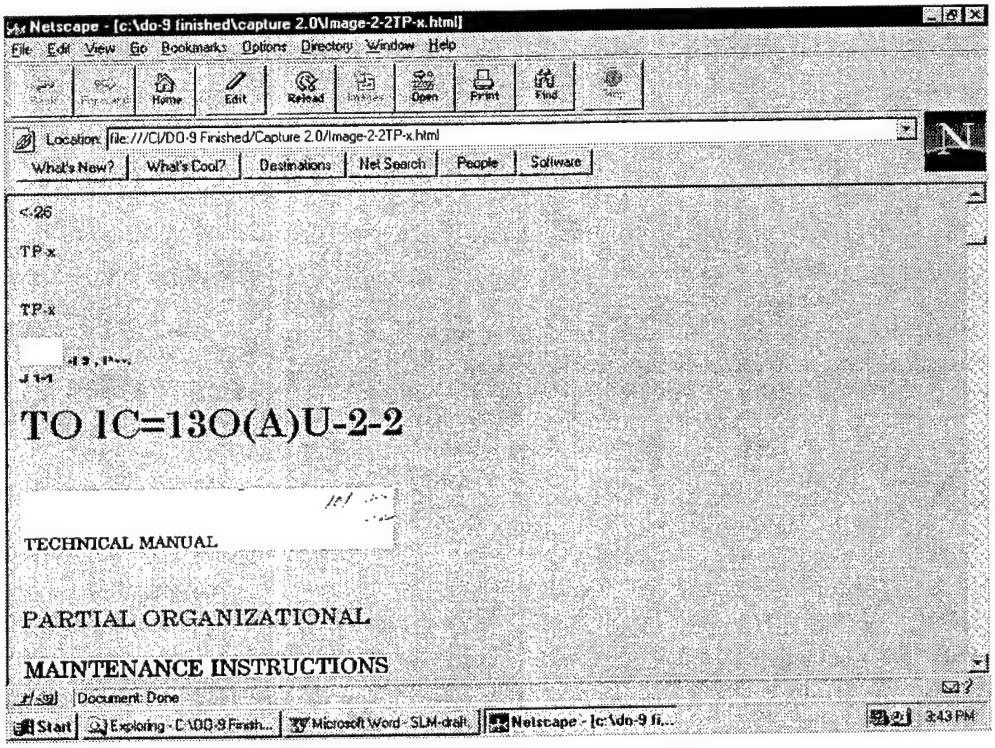


Figure 7-8. Adobe Capture Conversion to HTML Example

Caere OmniPage 8.0 also included the ability to convert to HTML. Figure 7-9 shows an example. The OmniPage HTML output was an improvement over Adobe's HTML output. OmniPage allows operator intervention prior to its OCR process to designate zones of text and graphics on a page. Although the graphics remained intact, OmniPage was unable to place the graphics where they belonged relative to the text on each page.

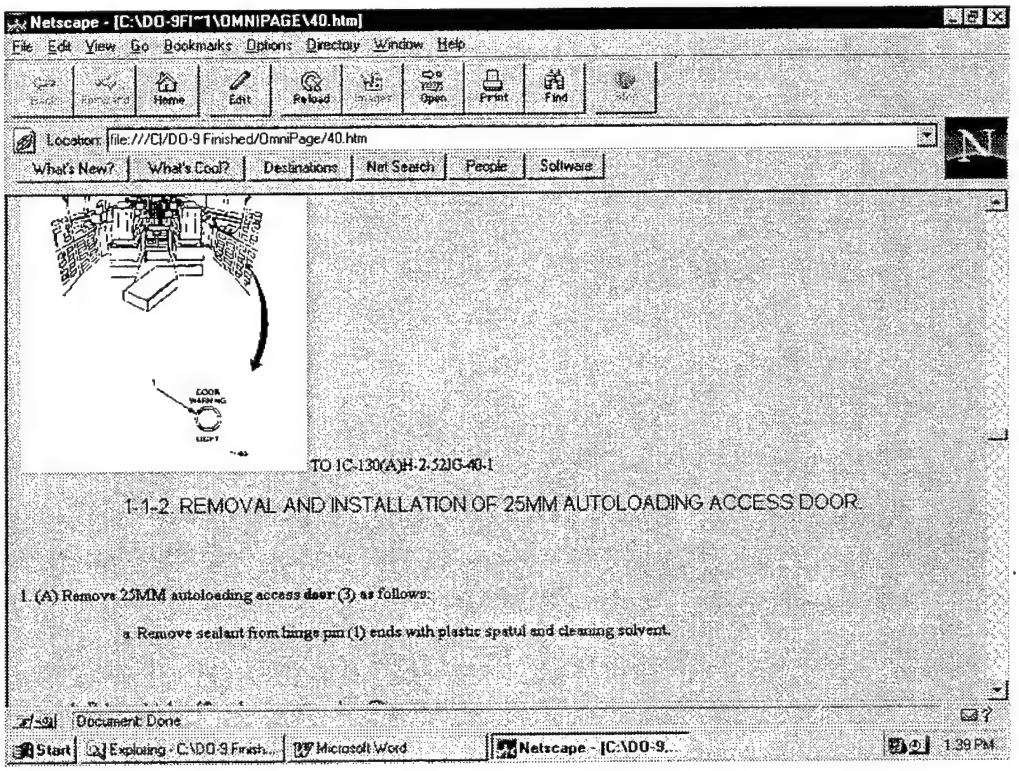


Figure 7-9. Caere OmniPage Conversion to HTML Example

Xerox ScanSoft TextBridge 98 was also evaluated because its upgrade included a conversion to HTML. The results were similar to the OmniPage output. As depicted in Table 7-5, a major weakness of TextBridge 98 is its inability to process volume. The software package quickly uses all of the available computing resources of the resident PC and produces unrecoverable errors after processing less than 40 pages. TextBridge 98 technical support suggested breaking the documents into 10 page or smaller files. This is not feasible for implementation.

There are other drawbacks aside from proofing and editing in using the HTML files produced by the OCR software packages. Formatting styles and pages break characters are lost. When viewed in Netscape, the HTML output scrolls as one continuous page, which makes navigation cumbersome. All links required within and between the HTML files must be inserted manually.

In Section 7.2, results of COTS package evaluations that take word processing formats and translate them to HTML format with the ability to automatically insert hyperlinks are

discussed. The MS Word format was chosen as the interim file format of choice to convert paper based information to HTML for the following reasons:

- The RTF and DOC files were usable from all four OCR software packages.
- Editing OCR processed documents is easiest in the word processing form.
- Maintaining technical information as a word processing document for configuration management is feasible.
- There are numerous commercially available software packages that can economically convert MS Word files to other electronic formats, including to HTML.
- Some TOs are readily available in MS Word format.
- Research indicated COTS packages are available to translate MS Word files to HTML with automatic hyperlink insertion.

7.1.5 Testing of OCR Software Packages

The MS Word file format was chosen as the best file type to convert paper based information to HTML. Each OCR package was evaluated further in order to determine which COTS OCR software would produce the best MS Word file.

Table 7-4 lists the criteria used in evaluating each package. A weight was applied to each set of criteria as some areas have more impact on the process and an effect on the conversion costs. Points were assigned for each feature/task category. Each feature/task is a part of the scan-OCR-proof-convert, edit/format process. The score was specific for Skill Level while the other criteria had variable score values from zero to five, with zero points scored for features that repeatedly failed or did not exist, and one to five points measuring performance, with five being the best. Weights were multiplied by the individual values and totaled for a final score. Final scores for each package are shown in Table 7-18. Tests were performed using 27 pages of TO 1C-l30(A)H-2-52JG-40-l, Technical Manual Job Guide, Organizational Maintenance, Structural Doors, Service Doors and 213 pages of TO 1C-130(A)H-2-71JG-00-1, Technical Manual Job Guide, Organizational Maintenance, Power Plant Operating Limits And Checklists. Each technical manual contained text, graphics, text within graphics, and tables. Seven wiring diagrams (11"x 17" page size) were also tested from TO 1C-l30(A)H-2-29GS-00-l, Supplemental Technical Manual, General System Organizational Maintenance, Hydraulic System. Each source document was a second-generation photocopy of original laser print quality manuals.

Table 7-4. OCR Package Evaluation Criteria

Criteria	Metric	Weight	Feature/Task	Value
Skill Level	Level of Experience	5	Student (High School or Higher)	9
			Administrative Personnel	5
			Computer Engineer	1
Speed	Seconds/Page	10	Import	0 to 5
	Pages/Minute		OCR Processing	0 to 5
	Pages/Minute		Conversion	0 to 5
Manual Intervention	Pages/Minute	20	Auto Process	0 to 5
			Auto Zone	0 to 5
			Manual Zone	0 to 5
			Training File	0 or 5
			Proofing	0 to 5
			Conversion	0 to 5
			Multiple Saves	0 to 5
Volume	Number of Pages	30	Volume Limitation	0 to 5
Usability of Output	Number of Errors	40	Character Errors	1 to 5
	Minutes/Document		Physical Formatting	1 to 5
	Graphics Embedding in Document Ability		Retain Graphics	1 to 5
	Yes or No?		Retain Format Styles	0 or 5

The criterion "Skill Level" was assigned a baseline weight of five. Skill Level was defined by the knowledge required for general use of the software package and category of personnel required operating the system. Skill Level was directly associated with the cost of operation. As depicted in Table 7-4, software that could be operated by a student (high school or higher) received a score of nine. Software that required operation by a level of administrative personnel received a score of five. Software that required operation by a computer engineer received as score of one. For each package, the learning curve was short; each package followed conventional MS Windows operating procedures, had easy to read user manuals, and on-line help.

Table 7-5. Rating Values for Speed

Criteria	Score	Definition
Speed	5	Fastest Speed
	4	Within 20% of Fastest Speed
	3	Within 21% to 49% of Fastest Speed
	2	50% or Lower than Fastest Speed
	1	Software failure during testing with a work around solution
	0	Software Packages lacking the feature being tested or software failures without a work around solution

Speed also effects the cost of operation and was weighted twice as heavily as Skill Level. Speed refers to the amount of time it takes to convert paper information to a usable MS Word file. The Speed criteria consisted of time to import the images into the OCR software and present the image page on the monitor to the operator, the time for the software to OCR each page, and the time to convert the OCR processed text to MS Word format. The time to scan the paper to images was equal as the paper was scanned one time and the same image file was imported into each OCR software package. For each area, Import, OCR Processing, and Conversion, the value for rating ranged from five to zero. The fasted package in each area scored five. The remaining packages were scored relative to the scores of four, three, and two. The value of one was reserved for a software failure during testing and zero was assigned to software packages without the ability being tested or multiple software failures without an alternate solution. Rating values are described in Table 7-5.

Manual Intervention included several Features/Tasks that are used by an operator to process the paper information to electronic format. This criteria was considered four times more important than the level of skill required to operate the package and two times as important as the speed of the package processing times. Like Speed, the Feature/Task criteria were assigned values ranging from five to zero. The sum of the scores was the end result for Manual Intervention in Table 7-18.

Auto Process was defined as the overall ability of the software to automatically import images by page (autoload), automatically determine the location of text and graphics on an image page (zoning), recognize the information (OCR), and convert the information (save as) to an MS Word file. Rating values for Auto Process are described in Table 7-6.

Table 7-6. Rating Values for Auto Process

Feature/Task	Score	Definition
Auto Process	5	Software can automatically process all four steps without manual intervention.
	4	Software can automatically process three of the steps without manual intervention.
	3	Software can automatically process two of the steps without manual intervention.
	2	Software can automatically process one of the steps without manual intervention.
	1	Software failed during testing with a work around solution.
	0	Software packages lacking the feature being tested or software failures without a work around solution.

Since the evaluation of Auto Process rated the overall ability of a software package to automate the combination of steps to convert paper to Ms Word without manual intervention, it was recognized that each step of the process could also be preformed individually. Auto Zone is the only step of Auto Process that had an effect on the converted file that is not relative to Speed. Auto Zone was defined as the capability to automatically determine the location of text and graphics on an image page (zoning). Rating values for Auto Zone are described in Table 7-7.

Table 7-7. Rating Values for Auto Zone

Feature/Task	Score	Definition
Auto Zone	5	Software package correctly differentiated text from graphics, and retained text as part of a graphic when applicable.
	4	Software package correctly differentiated text from graphics, but OCR processed text that should remain part of a graphic.
	3	Software package correctly differentiated text from graphics, but split a single graphic into several separate graphic files.
	2	Software package could differentiate text from graphics, but could not handle the complexity of a TO. Results depict OCR processed text out of sequence as well as a single graphic split into several separate graphic files.
	1	Software failed during testing with a work around solution.
	0	Software packages lacking the feature being tested or software failures without a work around solution.

The MS Word output results of the Auto Zone feature were similar to the results of Adobe Capture results to HTML displayed in Figure 7-8. It was determined that TOs containing text only could benefit from the automatic or no zoning features.

Manual Zone was defined as the ability to manually define the areas of text and graphics on a page. Manual zoning ensures the integrity of the figures as well as the flow of text. Being able to retain the zones from one page to the next page is advantageous for text and graphic areas that are consistent from page to page. As well, manual zoning allows for the elimination of external noise and markings (such as paper punch and staple holes or pencil markings in margins) to be removed before OCR. Rating values for Manual Zone are described in Table 7-8.

Table 7-8. Rating Values for Manual Zone

Feature/Task	Score	Definition
Manual Zone	5	Software package could automatically zone and zones could be edited individually to change size, definition (text, image, or table), or sequence and retained to the next image presented in the document. The page image could be rotated prior to zoning.
	4	Software package allowed for manual zones and zones could be edited individually to change size, definition (text, image, or table), and sequence. Zones retained to the next image presented in the document. Zone template could be saved and retrieved for later use (on same or different document). The page image could be rotated prior to zoning.
	3	Software package provided manual zone capability as in score "4" above but could not save a zone template.
	2	Software package provided manual zone capability as in score "4" above but could not maintain zones to next image presented in the document or save a zone template.
	1	Software failed during testing with a work around solution.
	0	Software packages lacking the feature being tested or software failures without a work around solution.

Training File was defined as the capability to create a separate data file with a bitmap character shape associated with the correct text character. Training files are usually created on a few pages of a large document and then loaded for the OCR of the entire document in order to improve character recognition accuracy. Improved character recognition results in less character errors. The ability to train OCR software packages on the associated characters allow for degenerated documents which once may have been considered unrecognizable to be OCR processed. Training data files can be saved and loaded for OCR processing documents with the same document characteristics (such as same font, same/like style, printed from the same/like

printer, or copied on the same/like copier). Rating values for Training File are described in Table 7-9.

Table 7-9. Rating Values for Training File

Feature/Task	Score	Definition
Training File	5	Software has ability to create and save a training file.
	0	Software packages lacking the feature being tested or software failures without a work around solution.

Proofing was defined as the ability for the operator to manually verify OCR processed characters against the bitmap images of the characters before the conversion to the chosen end result file. Options for proofing enable the operator to correct errors one time for multiple conversions and to correct errors more easily. Features in proofing/verifying include color-highlighted characters that fall below a predetermined confidence threshold, color highlighted words which do not match a predetermined lexicon, and the ability to check some formatting characteristics. Rating values for Proofing are described in Table 7-10.

Table 7-10. Rating Values for Proofing

Feature/Task	Score	Definition
Proofing	5	Software package proof or review has the following capabilities: • Allows end user to define level of accuracy confidence • highlights unrecognized characters • highlights unrecognized words checked in a user defined lexicon • presents the bitmap of the questionable character during proofing • allows editing of text format (bold, font, point size, underlines, tabs, spacing) • signals an image page or an image within a page • allows proofing per page before entire document is recognized • allows a save in proofing mode (to continue to edit later or convert a portion of a document)

Table 7-10. Rating Values for Proofing (Continued)

Feature/Task	Score	Definition
	4	Software package proof or review has the following capabilities: • Allows end user to define level of accuracy confidence • highlights unrecognized characters • highlights unrecognized words checked in a user defined lexicon • presents the bitmap of the questionable character during proofing • allows editing of text format (bold, font, point size, underlines, tabs, spacing) • allows cut and paste procedure • signals a text region • signals an image page or an image within a page
	3	Software package proof or review has the following capabilities: • Allows end user to define level of accuracy confidence • highlights unrecognized characters • highlights unrecognized words checked in a predefined lexicon
	2	Software package only highlights unrecognized characters after the entire document has been OCR processed.
	1	Software failed during testing with a work around solution.
	0	Software packages lacking the feature being tested or software failures without a work around solution.

Conversion was defined as the ability to control output page settings such as margins, fonts, hard page breaks, columniation, and retention of graphics as part of the MS Word file. The output format types were also evaluated in the Preliminary Evaluation of COTS OCR Software Packages. The speed of conversion was assessed in the criterion Speed. Rating values for Conversion are described in Table 7-11.

Table 7-11. Rating Values for Conversion

Feature/Task	Score	Definition
Conversion	5	Software package conversion has the following page setup capabilities: • control of output page size • control of margins • control of font (force to style and point size) • retain bold, italic, and underlined text • retain or eliminate hard page breaks • columniation • control of page justification • control of indentation • control of line spacing • retention of graphics within output file • control of graphic format • split files on blank pages (multiple documents in one OCR processing session)
	4	Software package conversion has the following page setup capabilities: • retain bold, italic, and underlined text • retain or eliminate hard page breaks • retention of graphics outside of output file • control of graphic format outside of output file
	3	Software package conversion has the following page setup capabilities: • retain bold, italic, and underlined text • retain or eliminate hard page breaks • no control of graphics
	2	Software package conversion automatically tries to convert the output file to retain original format to the best of its ability without operator controlled settings.
	1	Software failed during testing with a work around solution.
	0	Software packages lacking the feature being tested or software failures without a work around solution.

Multiple Saves was defined as the ability to convert (save as) a single document of OCR processed information (text and/or graphics) to multiple output formats. Rating values for Multiple Saves are described in Table 7-12.

Table 7-12. Rating Values for Multiple Saves

Feature/Task	Score	Definition
Multiple Saves	5	Software package has the following multiple save capabilities: • ability to save during OCR processing (i.e. to leave at the end of the day or use PC for other, unrelated tasks) without rescanning • ability to save a scanned document with specific OCR settings in order defer OCR processing (i.e. to OCR process many scanned documents during the night) • ability to convert the OCR processed document to another output format at a later time without rescanning • ability to convert graphics to individual graphic files at a later time without rescanning
	4	Software package has the following multiple save capabilities: • ability to save during OCR processing (i.e. to leave at the end of the day or use PC for other, unrelated tasks) without rescanning • ability to convert the OCR processed document to another output format at a later time without rescanning • ability to convert graphics to individual graphic files at a later time without rescanning
	3	Software package has the following multiple save capabilities: • ability to convert the OCR processed document to another output format at a later time without rescanning • ability to convert graphics to individual graphic files at a later time without rescanning
	2	Software package has ability to convert the OCR processed document to other output formats at a later time without rescanning.
	1	Software failed during testing with a work around solution.
	0	Software packages lacking the feature being tested or software failures without a work around solution.

Issues related to volume can paralyze an OCR process, causing bottlenecks, system crashing, rescans, and lost work. Volume has been weighted at 30 points. The number of pages that can be processed into one document file has a bearing on speed and manual intervention. Low cost COTS OCR software packages utilize resources of the computer workstation on which they are installed. The ability to process pages is dependent on the hardware resources. If the OCR software package uses large amounts of resources to process a few pages, errors and failures of the hardware and/or software will occur. Without some volume capability (100+ pages), the operator might have to split documents into small files of 10-20 pages. This requires more knowledge of technical information (higher skill level) to determine file breaks and how to organize information and creates additional manual intervention.

In addition to the system resources required for volume processing, some OCR software packages provide the ability to save a document in the OCR processed format prior to conversion. As mentioned in Table 7-12, storing this interim format file allows for OCR processing of additional pages to a document at a later time, converting to alternate and additional file types at a later time, and recovering work due to unforeseen computer hardware and software failures. Volume, also, was assigned values ranging from five to zero. Rating values for Volume are described in Table 7-13.

Table 7-13. Rating Values for Volume

Criteria	Score	Definition
Volume	5	Software packages with the ability to OCR process 212 pages without exhausting PC resources, degrading processing speed, or producing a volume limitation error message during conversion.
	4	Software packages with the ability to OCR process 212 pages without exhausting PC resources or degrading processing speed.
	3	Software packages with the ability to OCR process 100 pages without exhausting PC resources, degrading processing speed, or producing a volume limitation error message during conversion.
	2	Software packages with the ability to OCR process 100 pages without exhausting PC resources or degrading processing speed.
	1	Software failed during testing with a work around solution.
	0	Software packages lacking the feature being tested or software failures without a work around solution.

All of the costs associated with the OCR, proofing, and the conversion process would be better spent elsewhere if the output from the OCR package was unusable. For some degenerated paper originals, reconstructing the document would be more cost effective than scanning and OCR processing. Weight applied to the Usability of Output score was 40. Factors considered in determining usability were the number of individual character errors, time spent editing the physical format (cleaning up page layout), how well software packages retained graphics, and if the software packages retained formatting styles. Rating values for Usability of Output are described in Tables 7-14 to 7-17. The sum of the scores was the end result for Usability of Output in Table 7-18.

Character errors are can be corrected in the proofing editor or the final output. The number of errors per document quickly determines the usability of the document. It is possible to have an abundance of character errors on one page or several similar pages and few, if any errors on other pages. It is important to use to preserve individual character quality prior to scanning with paper preparation and prior to OCR processing by using the pre-OCR process Features/Tasks described in the Manual Intervention portion of the testing. Ratings values for character errors described in Table 7-14.

Table 7-14. Rating Values for Character Errors

Feature/Task	Score	Definition
Character Errors	5	Software packages producing output with an average less than 3% character errors.
	4	Software packages producing output with an average less than 5% character errors.
	3	Software packages producing output with an average less than 10% character errors.
	2	Software packages producing output with an average greater than 11% - 49% character errors.
	1	Software packages producing output with an average greater than 50% character errors.

Table 7-15. Rating Values for Physical Formatting

Feature/Task	Score	Definition
Physical Formatting	5	No time spent on MS Word output file for reformatting chapter headings, figure titles, margins, page breaks, and font.
	4	1 - 15 minutes or less spent on MS Word output file for reformatting chapter headings, figure titles, margins, page breaks, and font.
	3	15 – 30 minutes or less spent on MS Word output file for reformatting chapter headings, figure titles, margins, page breaks, and font.
	2	30 – 59 minutes or less spent on MS Word output file for reformatting chapter headings, figure titles, margins, page breaks, and font.
	1	1 hour or more on MS Word output file for reformatting chapter headings, figure titles, margins, page breaks, and font.

Table 7-16. Rating Values for Retain Graphics

Feature/Task	Score	Definition
Retain Graphics	5	MS Word output file retained all graphics. Each graphic was embedded in the appropriate page between the appropriate text as in the original document. Each graphic is an MS Word picture, editable in MS Word.
	4	MS Word output file retained all graphics. Each graphic was embedded at the bottom of each appropriate page. Each graphic is an MS Word picture, editable in MS Word.
	3	MS Word output file retained all graphics. All graphics were placed at the end of the text in the output file.
	2	MS Word output file retained no graphics. All graphic images were converted to individual files. Each graphic required manual insertion into the MS Word file.
	1	MS Word output file retained no graphics and the graphics converted were not able to be imported into the MS Word file.

Format styles refers to the headings, paragraphs, figure titles, etc. defined and referenced for blocks of text in word processing packages. Retaining the format style in an OCR processed document requires the software to define set of style characteristics for a text region (indentation, justification, font, point size, etc.). For each text region recognized, the OCR software tags the

block of text with the appropriate style. Each text region tagged with a style is recognized by word processing packages as such.

Table 7-17. Rating Values for Retain Format Styles

Feature/Task	Score	Definition
Retain Format Styles	5	Software package has the ability to define and maintain style for like text regions within a document.
	0	Software packages lacking the feature being tested or software failures without a work around solution.

Table 7-18. Results of OCR Software Package Evaluation

	Adobe Acrobat Capture 2.0	Caere OmniPage 8.0	Caere WordScan Plus 4.0	Xerox TextBridge 98
Skill Level	9	9	9	9
Speed	70	70	110	100
Manual Intervention	140	400	400	440
Volume	60	90	120	0
Usability of Output	240	320	400	560
TOTAL	519	889	**1039**	1109
PRICE	$895 + KEY*	$499	**$595****	$79

*Adobe Acrobat Capture 2.0 requires a hardware key counter attached to the processing workstation's parallel port. Each page processed is counted as a charge against the key. The first key included in the $895 price is valued at 20,000 clicks. Additional keys are sold for $595 for 20,000 more pages and $4,995 for 200,000 pages.

**Caere WordScan Plus 4.0 can be purchased for less in software retail stores.

Caere WordScan Plus 4.0 scored the highest number of points as listed in Table 7-18 above. Features include:

- Previewing, zoning (up to 99 areas of images or text - manual), deskewing (straightening images from crooked scans) and verifying before conversion to MS Word
- Single- or multi-page processing
- Immediate or unattended processing
- Saves recognized but unconverted files
- Forward and backward image verification for on screen text proofing
- OCR pages with mixed images and text and save into a single file.

7.2 ELECTRONIC FORMAT CONVERSION TO HTML & LINKING

7.2.1 HTML Overview

HTML is a simple markup language used to create hypertext documents that are portable from one platform to another. HTML documents are SGML documents with generic semantics that are appropriate for representing information from a wide range of applications. In this study we focused on producing output that was compliant with W3C's HTML 3.2 specification. The HTML 3.2 specification provides the capability of creating hyperlinks in a document. A hyperlink provides a way to rapidly move to different locations in the current document or to jump to an entirely different document. The functionality available in HTML 3.2 is nearly equivalent to the functionality available in a class 3 IETM.

7.2.2 Preliminary COTS Evaluation of Translation to HTML

There are many advantages for producing TOs in HTML. One advantage is that the TOs can be viewed using no or low cost HTML viewers such as Netscape's Navigator or Microsoft's Internet Explorer. Another advantage of using HTML is that it is an open standard. As a result of being an open standard, the HTML specification is publicly available and many HTML viewers exist.

During this phase of the effort, the team did an evaluation of the four products for converting electronic documents to HTML mentioned in the Technology Assessment section. The four products that were evaluated and the companies that produced each product are listed Table 7-19.

Table 7-19. HTML Conversion Packages

Product	Company
ANT	Shareware
Easy Help/Web	Eon Solutions Ltd.
Web Publisher	Ski Soft
HTML Transit	Info Access

Each of the above packages were evaluated to determine the best product for converting the electronic documents into HTML.

The following criteria identified in Table 7-20 were used to determine which of the four candidate packages would be best suited for use in converting to HTML.

Table 7-20. Evaluation Criterion

Criteria	Value(s)
Cost	Cost of package
Auto Linking	Yes/No
Input Files	Acceptable Input Files
Output HTML 3.2	Yes/No
Manual Intervention	Extensive/Minimal
Pattern Matching	Yes/No

As mentioned in the Technology Assessment section, the overall software budget for the study was $2000. The "Auto Linking" criterion was one of the primary features that the package needed to support. The term "Auto Linking" was defined to mean the capability to intelligently insert hyperlinks into the document for easy navigation within the document. The ability to do "Auto Linking" was one of the main differentiating factors among the packages evaluated. At the time of the evaluation of the conversion packages, it was assumed that either an MS Word file format and/or HTML would be used as an input file of choice for the conversion. As a result of this requirement, the conversion application needed at a minimum, to accept an MS Word or HTML files as input. The output from the conversion needed to be compliant with the HTML 3.2 standard. Throughout the study, the team concentrated on determining a solution that would need a minimal amount of manual intervention. The manual intervention criterion was used to evaluate how much manual intervention was needed each time a document was converted. The final criterion titled "Pattern Matching" was defined as the ability to recognize patterns in the input document and format the results of the matched pattern in the resulting HTML. An example of pattern matching capability would be for the conversion application to identify all occurrences of the word "Figure" followed by any number of digits separated by a period. Examples of the mentioned pattern would include: "Figure 1", "Figure 1.2", "Figure 1.2.3". The only package that had pattern matching capabilities implemented this feature using regular expression syntax.

In this section, the results from the evaluation of the four translation packages are presented. Each of the packages will be identified followed by how the package scored against the evaluation criteria. At the end of the section, a table summarizes the results of the evaluation as well as identifies the package that was chosen as the best candidate conversion application. Evaluation copies for each of the four packages were obtained and loaded on a personal computer with a 120 MHz Pentium Processor and 64 MB of RAM. A sample TO was used to test each of the packages. This sample document contained text, tables, and graphics typical characteristics of the majority of the TOs examined. Since all of the packages reviewed accepted MS Word file format as an input type, the same Word file was used to evaluate each of the packages.

7.2.2.1. Evaluation of ANT

The ANT software package was a template file for MS Word. This package extended the functionality of the MS Word program and allowed the user to easily convert existing Word documents to HTML. As a result of being an MS Word template file, it was able to handle MS Word files as an input type. The tool did an adequate job of converting the TO to HTML, but it exhibited trouble deciphering the different cases and sizes of the characters. The package was not able to maintain a consistent look and feel throughout the HTML document, which resulted in having to do a substantial amount of manual clean up to make the resulting document look acceptable. The ANT package lacked the capability to do any "Auto Linking" and had no pattern matching capabilities. The tool seemed to be designed to convert fairly simple Word documents into a single HTML file.

7.2.2.2. Evaluation of Eon Solutions Ltd., Easy Help/Web

Easy Help/Web, like ANT, was a program that extends the functionality of MS Word. This product exhibited capabilities similar to ANT. The Easy Help/Web product was primarily suited for converting small simple Word documents into a single HTML file. The processing speed of converting the file was slow. This tool lacked the capability to do any "Auto Linking" and had no pattern matching capabilities.

7.2.2.3. Evaluation of Web Publisher

Web Publisher was the first product that proved to be a viable candidate for converting the sample TO to HTML. This tool comes in both a standard and a professional version. The main difference, other than price, between the two versions is that the professional version allows the resulting output to be separated into multiple HTML files. This was an important feature due to the fact that the majority of the TOs are over 100 pages in length. The price of the standard version of Web Publisher at the time of the study was $495 and the professional version retailed at $990. The professional version of Web Publisher met all of the selection criteria except for the support of pattern matching capabilities. Web Publisher was able to take as input files both MS Word files as well as Word Perfect files. Automatic creation of a TOC through the use of hyperlinks was fairly easy to accomplish with the sample TO used for the evaluation. When the sample word document was loaded, Web Publisher identified all of the different styles contained in the document. Once the styles had been identified, the user could customize the resulting HTML based on the recognized styles. Web Publisher uses a concept of templates, which allows the user to format the resulting HTML based on the character and paragraph styling found in the source document. Templates can be created and stored, then reused on other documents. This proved to be a powerful feature for purposes of converting multiple similar documents. Once a template has been created for a type of document, it was simple to reuse this template for similar documents. Overall, this product was a solid candidate for converting the TOs to HTML, but the tool lacked pattern matching capability, and it did not accept HTML as an input file type.

7.2.2.4. Evaluation of HTML Transit

HTML Transit was a very complete conversion package sharing similar capabilities of the Web Publisher product. The retail price for the HTML Transit product during the time of the study was $495, which included 60 days of technical support. HTML Transit supports multiple input file types including MS Word, Word Perfect and HTML files. As with Web Publisher, HTML Transit identifies the styles that are present in the document, and allows the user to customize the resulting HTML based on each of the styles found in the source document. HTML Transit had a concept of templates, which could be saved and reused for multiple documents. Automatic generation of a table of contents (both global and local), and a list of figures are simple tasks to accomplish with the HTML Transit product. A Global TOC applies to the entire publication (which could consist of multiple documents) where a local TOC applies only to a particular HTML page. Another power feature in HTML Transit was the ability to match patterns by using regular expression syntax. The user could construct a pattern for which to search in the document, and then apply HTML styling for each occurrence of the identified pattern. This conversion software proved to be very powerful, but the learning curve may be a little longer than most due to the complexity of setting up the original template. Once the template has been completed, the conversion process becomes extremely simple to complete. Large documents (100 pages) converted through the template in less than 3 minutes. The only drawback to this software was that if links are desired between documents, they needed to be inserted manually. Automatic links can be established between the global TOC, local TOC, index, and body of a single source document.

7.2.2.5. Summary of Evaluation

Of the four packages evaluated, there were only two that proved robust enough to sufficiently handle the conversion of the test TO. The two viable packages, HTML Transit and Web Publisher, exhibited similar capabilities, but a few characteristics of the HTML Transit product propelled it to be the best package for this effort. One distinguishing characteristic of the HTML Transit product was price. HTML Transit's price was approximately $500 less then Web Publisher. Second, HTML Transit had an added feature of pattern matching capabilities not present in the Web Publisher product. Finally, HTML Transit provided much more functionality for customizing the resulting HTML. Table 7-21 summarizes the results of the evaluation of the HTML conversion packages.

Table 7-21. Final Results of HTML Conversion Packages

	HTML Transit	ANT	Easy Help	Web Publisher
Cost	$495	$39	$280	$990
"Auto Linking"	Yes	No	No	Yes
Input Files	Many	Word	Word	Many
HTML 3.2	Yes	Yes	Yes	Yes
Manual Intervention	Minimal	Extensive	Extensive	Minimal
Pattern Matching	Yes	No	No	No

7.2.3 Further Testing Translation using HTML Transit

In this section, a general overview of how HTML Transit works and a description of its features is given. Following the overview section will be a description of how HTML Transit was utilized to convert a sample TO.

Templates were the key to understanding how HTML Transit works. They can be thought of as style sheets for the resulting HTML documents. Word processing documents provide character styles like **bold**, underline, *italic*, etc. for formatting specific words or groups of words and paragraph styles like Heading 1, Heading 2, Body Text, etc. for formatting whole paragraphs. Paragraph styles are used not only to apply character formatting and line spacing information in a consistent way throughout the document, but also define the logical structure or outline of the document. For example, in a TO, all chapters might be defined using style Heading 1, and all chapter sections could be identified using style Heading 2.

The equivalent to a style or pattern in HTML Transit is called a Transit element. Creating a template consists of establishing the rules about how styles and selected formatting patterns in the source document are mapped to Transit elements, and how Transit elements are then mapped to HTML formatting. HTML Transit automatically creates Transit elements for all styles defined in the source document. For example, when HTML Transit loads a file that contains a Heading 1 paragraph style, a corresponding Transit element will be created. Formatting characteristics can then be applied to the Transit element, which will be present in the resulting HTML files.

Transit elements are also used to define the document structure and navigation. Users may select Transit elements to be included in reference pages for the document, such as a TOC, list of figures, list of tables, or keyword index. For example, all the Heading 1, Heading 2, and Heading 3 Transit elements might be included in the TOC, each hyperlinked to the content body. Users may also elect to split the document at every occurrence of a specified Transit element. This allows the user to break the document up into multiple HTML pages, making it faster to load and easier to navigate than a long single HTML document.

7.2.4 Example of Converting TO using HTML Transit

In this section, a detailed description is presented which describes how the sample TO was converted using HTML Transit. It is assumed that the particular style TO being used does not have an existing HTML Transit template file. Another assumption is that the example TO had been converted to an MS Word file and contains paragraph styles. The following styles are present in the document: Heading 1, Heading 2, ..., Heading 1 Warning. The Heading 1 style was used to identify chapters in the TO. Subsections within the chapters were styled with subsequent heading numbers beginning with Heading 2. For example, the title for section 1-1 was styled using Heading 2 and the title for section 1-1-2 was styled using Heading 3, etc. Also, if a particular section contained a Warning, Caution or Note, then a user-defined style was used to identify the title of the special condition. For example, if section 1-1 had a Warning, then the title for section 1-1 was styled using "Heading 1 Warning."

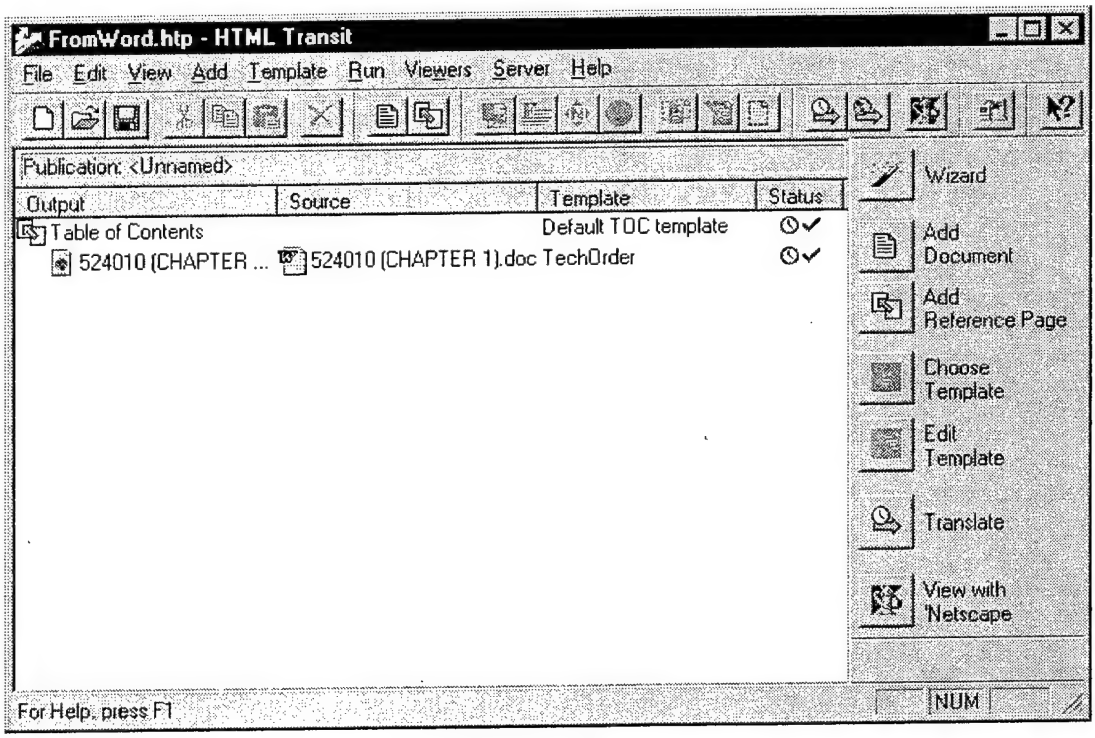


Figure 7-10. HTML Transit Main Window

Once HTML Transit has been loaded (Figure 7-10), the user needs to create a new publication. The publication maintains which source files are being used as well as the associated templates for each source file. Once the publication has been created, the desired documents that are to be converted to HTML are loaded. In this case, a single Word file containing the entire TO was loaded. As the file is loaded, HTML Transit identifies all the different styles present in the document and creates a Transit Element for each of the styles present. After the file has been loaded, the user will be presented with a preview of how the document will look after converted to HTML using a predefined template. Since a template for this type TO document had not been created, a new template was created. The newly created template had default styles for each of the Transit Elements created. These elements were modified in order to produce the desired output. Figure 7-11 shows all of the styles identified in the sample document.

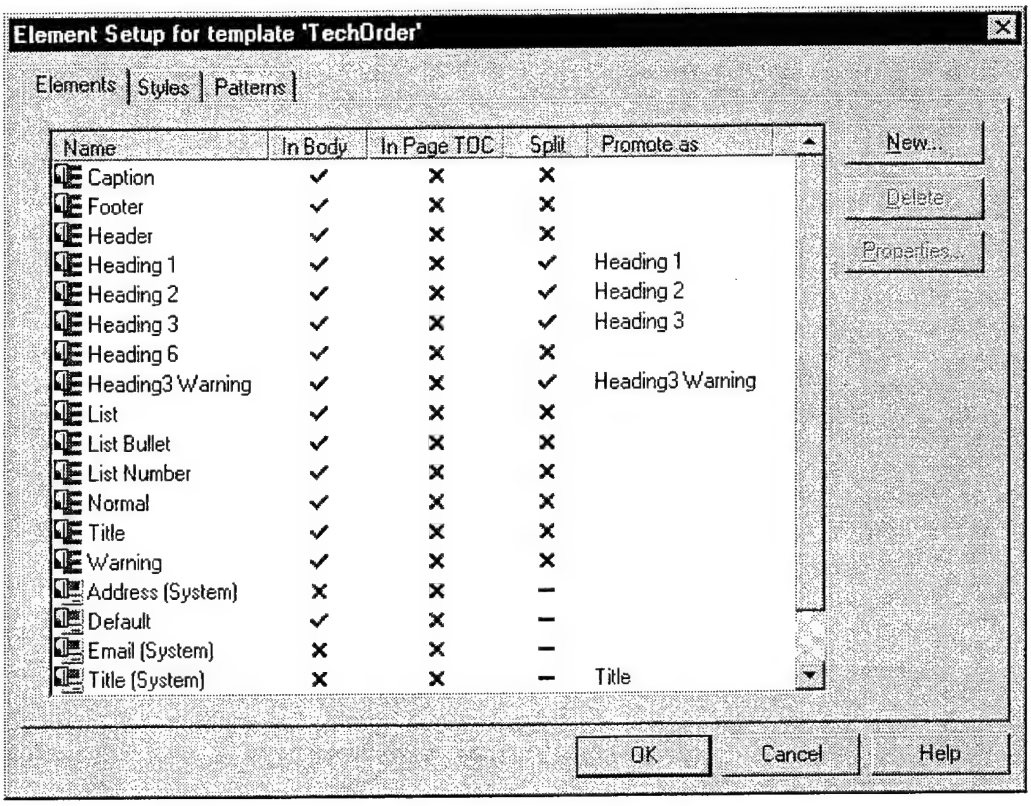


Figure 7-11. HTML Transit Element Setup

The next step in the conversion process is to add a reference page (Figure 7-10). This is accomplished by selecting the Add Reference Page option from the main window. In this example, we elected to create a TOC for the conversion. The main window now shows that the publication contains a TOC document as well as the TO document (Figure 7-10). At this point, the source document has been loaded, and a TOC has been created for the publication. The remainder of the steps needed to complete the conversion involves formatting the look and feel of the generated HTML by applying formatting properties to each of the Transit Elements. Both the TOC and the TO have their own set of Transit Elements. The TOC receives its Transit Elements from the source document as a result of the user selecting which Transit Elements from the source document should be present in the TOC's Transit Elements.

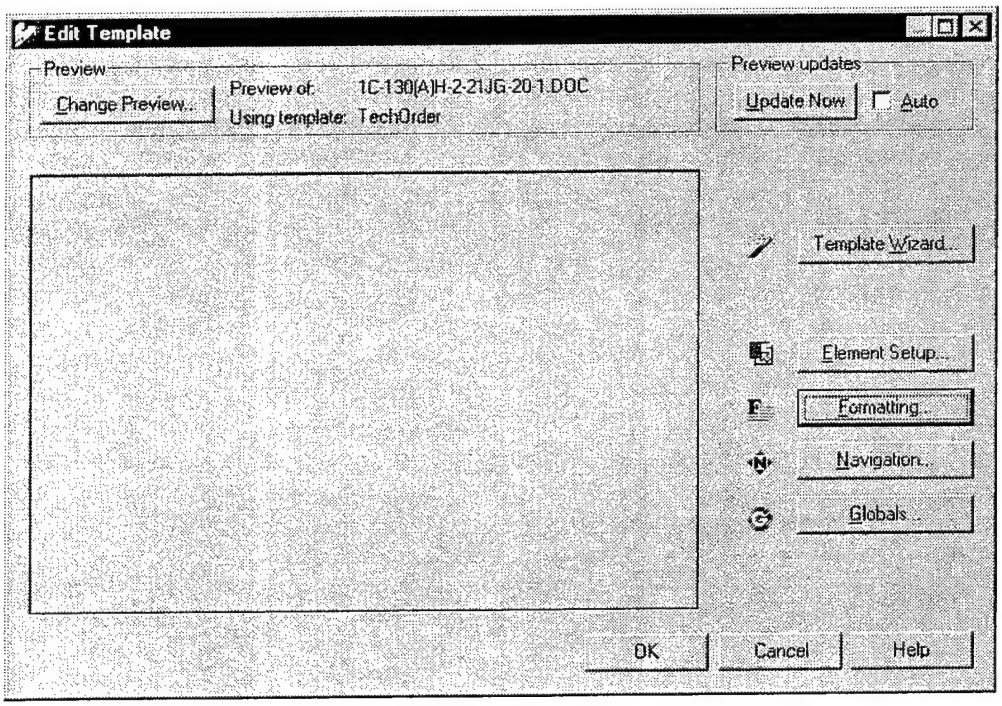


Figure 7-12. HTML Transit Edit Template

In HTML Transit there are four main categories for formatting the resulting HTML: Element Setup, Formatting, Navigation, and Globals (Figure 7-12). Element Setup formatting allows the user to configure four different aspects of the resulting HTML based on the Transit Elements (Figure 7-11). The first aspect that can be configured is selecting whether the Transit Element should be included or excluded from the HTML files. The decision of whether to include or exclude the Transit Element is indicated by a check mark "√" (yes) or an "X" (no). The next aspect of Element Setup formatting is selecting whether or not each of the Transit Elements should be included in the page TOC. For this example, none of the Transit Elements were selected to be included in the page TOC. In this example, it was decided to create a separate TOC, which will appear in a separate frame. Splitting the document into separate HTML files is accomplished by selecting the "Split" option for each Transit Element. As shown in Figure 7-11, it was decided to split the resulting HTML files for each occurrence of Heading 1, Heading 2, Heading 3 and Heading 3 Warning. Finally, in the Element Setup formatting section, the user can elect which Transit Elements should be promoted to the reference pages. In this example, Heading 1, Heading 2, Heading 3, Heading 3 Warning were selected to be made available to reference pages. As a result of promoting these elements, they will be present as Transit Elements in the TOC.

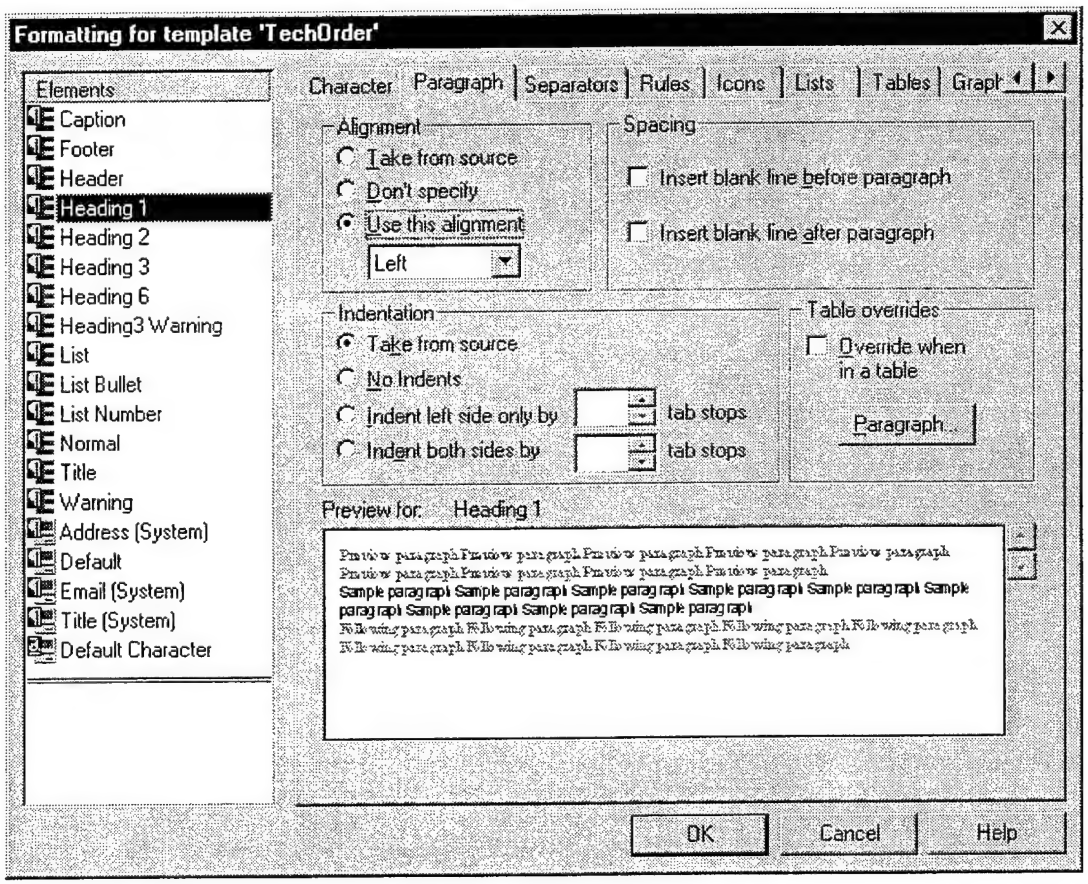


Figure 7-13. HTML Transit Element Formatting

The next category of customizing the resulting HTML is called Formatting. This allows the user to customize the generated HTML based on each of the Transit Elements. For each of the Transit Elements, the user is presented with a multiple tabbed dialog to customize the HTML. As shown in Figure 7-13, we elected to align each occurrence of the Heading 1 Transit Element to be left justified and to insert a blank line after each occurrence. There are many other formatting characteristics which can be set on a per Transit Element basis in this section. The user can also elect to supply his/her own HTML code to be included before or after each of the elements. This capability was utilized to format the Heading 3 Warning Transit Element in the TOC. HTML code was inserted to place the character "W" using a bold typeface and red font color in order to identify in the TOC that this section of the manual contained a Warning.

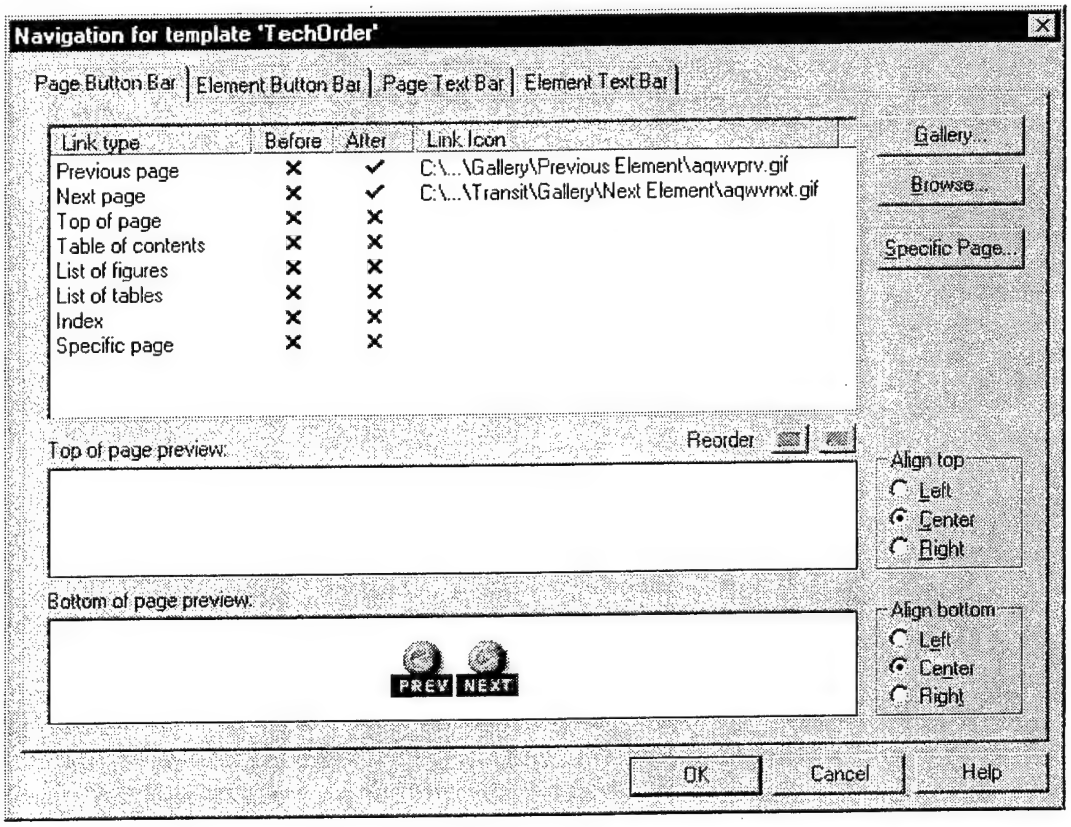


Figure 7-14. HTML Transit Navigation Formatting

The Navigation section of formatting allows the user to include navigational capabilities to the generated HTML pages. As shown in Figure 7-14, "PREV" and "NEXT" navigation icons were included in each of the separate HTML source files. This allows the user to quickly navigate to the next or previous page by selecting the appropriate icon. The inclusion of navigation icons at the bottom of each page was accomplished by selecting the "Previous Page" and "Next Page" link type from the Navigation window. The user can select what type of icon to use from the gallery of supplied icons or by using a user defined icon.

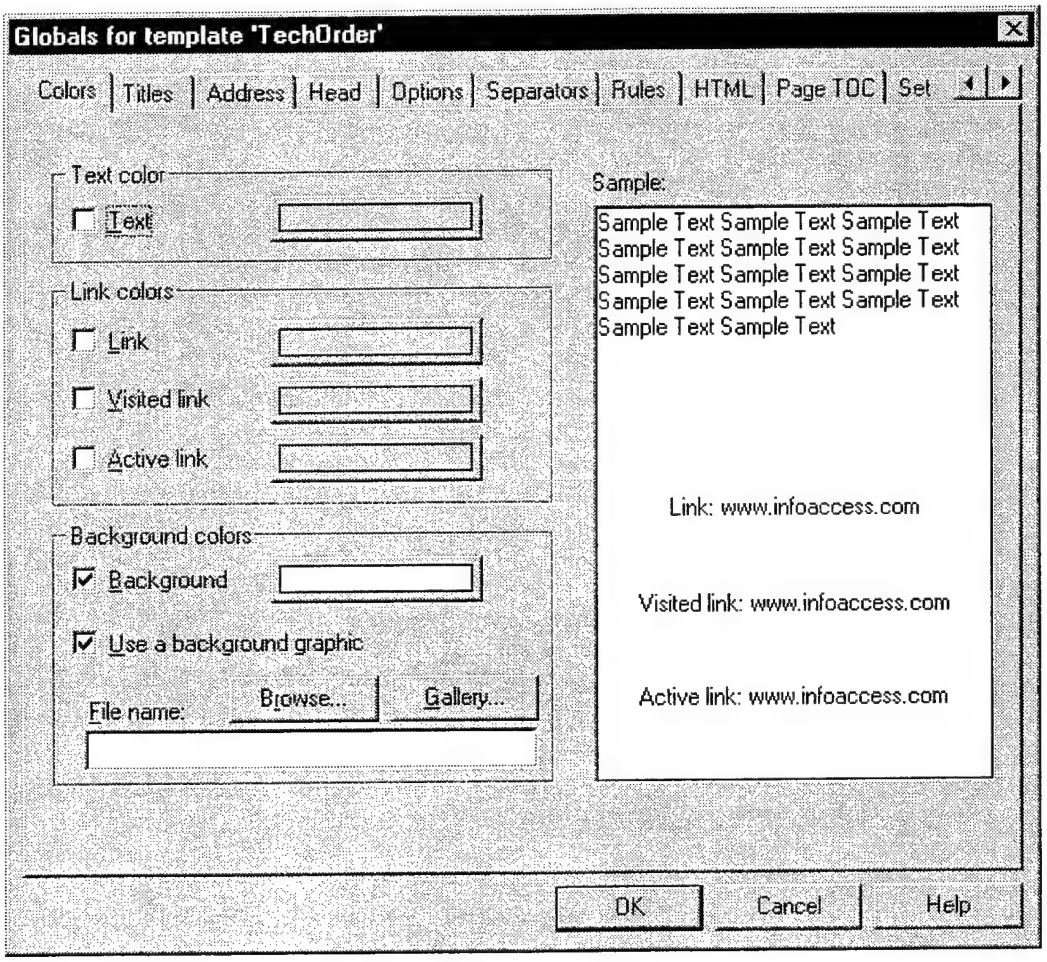


Figure 7-15. HTML Transit Global Formatting

Finally, global formatting of the document is accomplished by selecting the Globals button from the Edit Template window. The global formatting allows the user to customize characteristics such as background color and link colors for each of the generated pages (Figure 7-15). We elected to globally set the background color white for each of the generated pages. Another feature available in the global formatting section is the ability to insert user defined HTML for each of the pages. This feature was exploited to include a JavaScript function, which was used to provide some additional navigation capability.

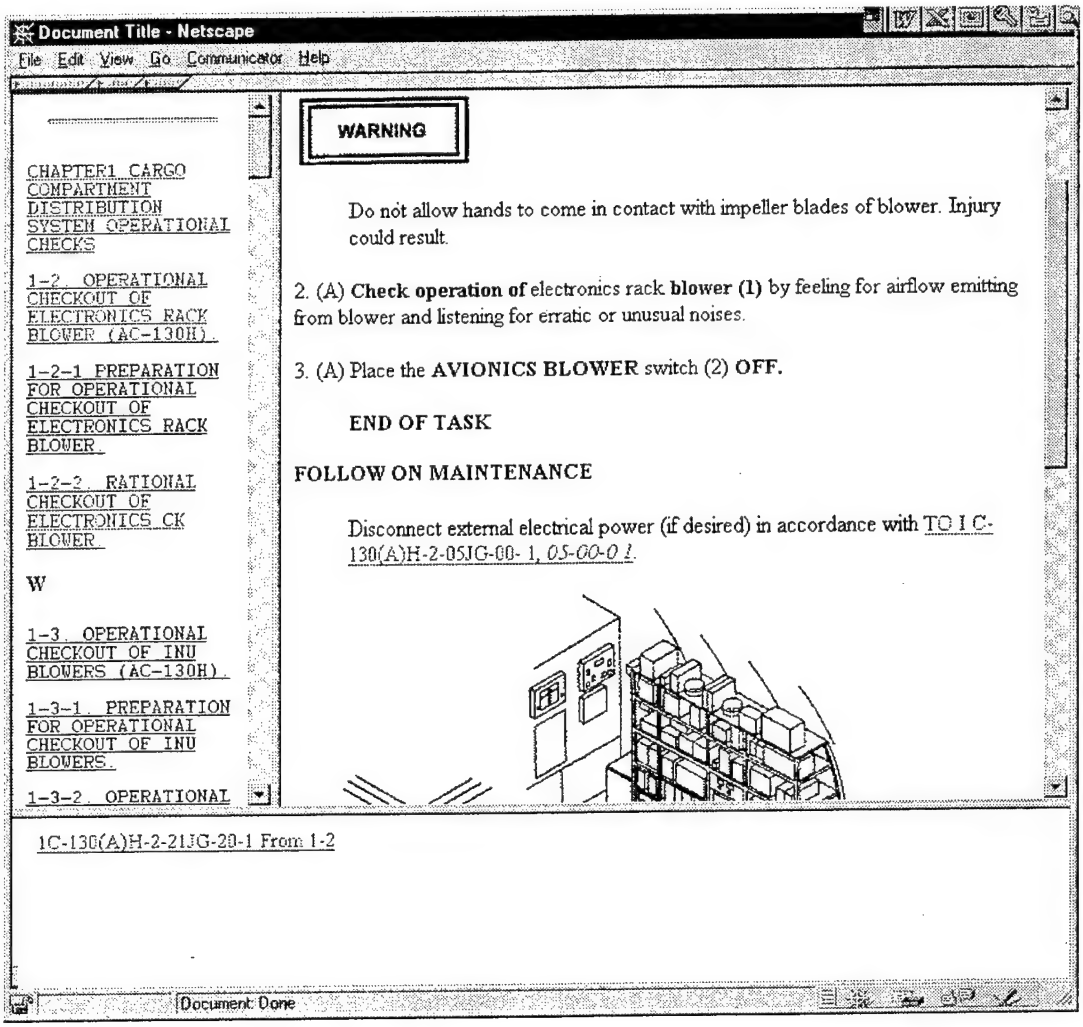


Figure 7-16. HTML Conversion Example Page

The four formatting sections described above allow the user to customize the template. As the user changes different formatting characteristics of the template, he/she can select the preview option in order to see the output. Once all the formatting has been completed for the publication, the translate option is executed which will translate the document and create the HTML files. Once satisfied with the resulting HTML, the user can elect to export the template so it can be used with subsequent documents. In this example, the template was exported and named TechOrder. Figure 7-16 shows the results of the translation from this TO. We elected to utilize frames in order to display the TOC in a separate frame. The frame at the bottom of the page was used to provide additional navigational capabilities, which will be described in more detail in a latter section.

After converting the sample TO, a meeting was conducted with government human factor specialists in order to demonstrate the results of the translation. Three main issues were identified which the specialists felt needed to be addressed. The first concern was the colors used to identify hyperlinks and traversed hyperlinks. The specialist remarked that the colors used for

the hyperlinks and traversed hyperlinks were not ideal. After investigating this issue, it was found that the hyperlink and traversed hyperlink colors could be set in the generated HTML using a Global setting in HTML Transit. The only potential issue with this solution is that the user could configure his/her browser such that hyperlink colors identified in the HTML are ignored, and the colors selected by the user are used. It was determined that if the user had configured his/her browser to ignore link colors that he/she had made a conscious decision to select the desired link colors and that these colors should be appropriate for the user.

The next issue that was identified by the specialist was the identification of Warnings, Notes, and Cautions. It was noted that as a result of the TO being presented in HTML, the user could rapidly navigate to a particular section of the document. The concern with this navigational capability was that the maintainer might jump to a section and disregard that a previous section might contain a Warning, Caution or Note. The solution to this was, for each section in the TO containing a Warning, Caution or Note, to identify each of these sections in the TOC with a special icon. The special icon in the TOC would serve to notify the maintainer that the section contained a Warning, Caution or Note.

The final issue identified by the specialists was a concern that the maintainer could be easily confused when jumping from one TO to another. The main issue with jumping to another referenced manual was that the user needed a clean way to navigate back to the specific section of the original document from which he/she selected the hyperlink. By utilizing JavaScript, a solution was developed to aid the user when navigating between different TOs. Each time the user selected a hyperlink, which referenced a different TO, the page, and position within the page was recorded. A hyperlink was generated in a separate frame, that when selected would navigate the browser back to where the hyperlink was selected.

7.2.5 Conversion Process from SGML to HTML

Since there are currently contractors delivering SGML tagged data to the USAF, the TASC/TAMSCO team researched ways to convert the SGML files to HTML. There were two products found that would convert SGML files into an HTML format. Those products are the ArborText product suite which costs about $5000 and the Adobe FrameMaker + SGML which costs about $2000. TAMSCO owns the ArborText product suite, so there was a complete set of software. The Adobe product was an evaluation copy.

Although the ArborText product suite is more expensive, it is more recognized in the industry than Adobe FrameMaker + SGML. The industry also perceives the ArborText suite of products to be more reliable at this point in time. Compared to ArborText's Adept Editor, Adobe FrameMaker + SGML is a relatively new product.

ArborText, Inc. and Adobe have been in business for sixteen years. Market research has shown that, of all SGML authoring seats sold, a third are ArborText. If small businesses and academic institutions are taken out, that number jumps to two-thirds. ArborText appears to be outselling Adobe FrameMaker by more than five times. ArborText confirmed that this is their belief. Adobe did not respond to requests for this type of information.

USAF DTDs that the AF PDSM office produce utilize an SGML construct #CONREF that is not supported in Adobe FrameMaker + SGML, but is supported by the ArborText product suite. ArborText has implemented "native" SGML. This means they use the ISO standard as is. Adobe uses filtered SGML, but can output a valid or invalid SGML file.

Both products can save SGML files out to multiple file formats such as ASCII and HTML. The version of HTML they have chosen to support is version 3.0, not 3.2.

Both products are customizable for the end user. Adobe requires customization just to accept the USAF DTDs. This proved to be too time consuming for this study. ArborText will accept the USAF DTDs with or without customizing the product.

Since the USAF DTDs could not be brought into the Adobe product, the SGML to HTML procedure that follows was done using the ArborText Adept series of products. The actual SGML to HTML conversion took place on a Sparc20 with Solaris 2.5.

To convert the SGML files to HTML, the process is relatively simplistic. As long as the SGML tagged instance is structurally valid, the process is just a simple "Save As" procedure. If the document is not structurally valid, meaning it does not parse, then the document must be brought into compliance with the DTD.

After the document is valid and the "Save As" procedure has been selected, the file format must be chosen. When choosing HTML, if a mapping file called the .q20 file does not exist, the software creates one. After viewing the results of the conversion, the .q20 file may or may not be modified to use again.

The .q20 file lists parts of the HTML tag set. The SGML tag names are placed under the appropriate HTML tag. There is also an area to omit SGML tags from the conversion. An example of when you would want to do this is as follows:

Suppose a portion of the tagged instance has the following markup:

<para0><title>Scope</title>
<para>The scope of this particular section is to discuss the SGML to HTML conversion using the ArborText Adept series product suite.</para></para0>

In the .q20 file the <title> could be mapped to a heading and the <para> to a paragraph. There is no need to convert the <para0> tag. It would be put in the "omit" area, and the <title> and <para> would still be converted to HTML.

If modifications are made to the .q20 file, after it is saved, the original SGML file can be "Saved As" again. After checking the results of the second conversion if it is acceptable, other SGML valid documents that have been tagged to this particular DTD may be "Saved As" HTML without having to modify the .q20 file.

8.0 SCENARIO AND CONCEPT DEMONSTRATION

A maintenance task scenario was presented to the COPIS team, by AFRL/HESR in order to test the process and quantify the costs for converting paper TOs to HTML. The scenario consisted of three TOs totaling 485 pages. Three separate processes were used to create HTML files of technical information capable of being viewed via Netscape: 1) Paper to MS Word and MS Word to HTML, 2) MS Word to HTML, 3) SGML to HTML.

8.1 PAPER TO MS WORD AND MS WORD TO HTML

A demonstration of the scenario as HTML files viewed through Netscape was constructed using Caere WordScan Plus 4.0 for converting paper to MS Word files and InfoAccess HTML Transit 4.0 for translating the MS Word files to HTML. The hyperlinks and JavaScript for references within a TO were automatically inserted by Transit into the HTML file. References to external TOs were hyperlinked manually. Each source document for the scenario is listed in Table 8-1 and was a second-generation photocopy of original laser print quality manuals. Each contained text, graphics, text within graphics, and tables. Costs associated with the process in Figure 8-1 can be found in Table 9-1.

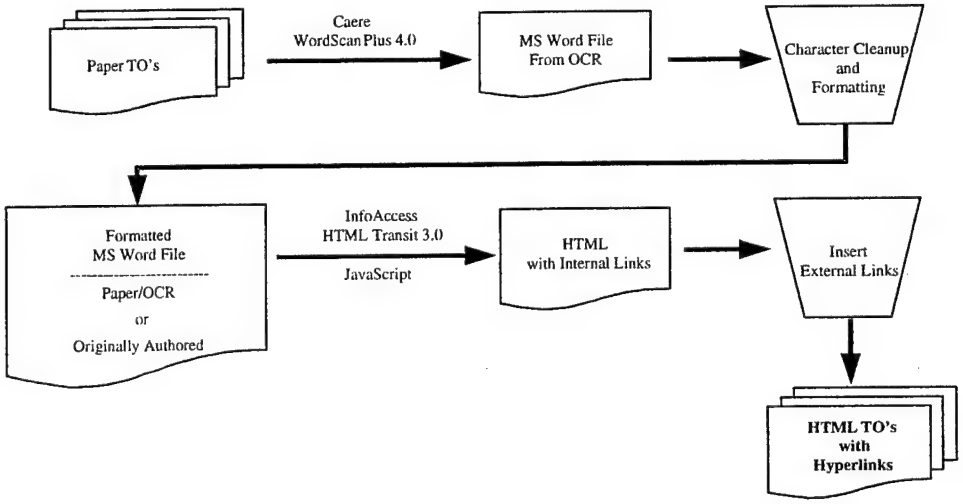


Figure 8-1. Conversion Process

The process for TOs originally authored in MS Word is depicted in Figure 8-1 starting with the box titled "MS Word File From OCR". The TO, listed in Table 8-2, used to demonstrate the MS Word to HTML conversion is similar to the MS Word file of its paper counterpart which was scanned and OCR processed to MS Word. Since all TOs currently in MS Word format do not model the style required for the HTML Transit template, costs for formatting the MS Word files have been calculated. These costs are found in Table 9-2.

Table 8-1. Paper Source Document Information

TO Number	TO Name	# of Pages	Time to Convert to HTML
TO 1C-130(A)H-2-71JG-00-1	Technical Manual Job Guide, Organizational Maintenance, Power Plant Operating Limits and Checklists	213	15.75 hours from paper
TO 1C-130(A)H-2-21JG-20-1	Supplemental Technical Manual Job Guide, Organizational Maintenance, Air Conditioning Distribution System	170	12.93 hours from paper
TO 1C-130(A)H-2-05JG-00-1	Supplemental Technical Manual Job Guide, Organizational Maintenance, Ground Handling General Maintenance	102	5.52 hours from paper

* The wiring diagrams were not OCRed but captured during the scan process as image only at 300 dpi to reduce file size.

Table 8-2. MS Word Format Source Document Information

TO Number	TO Name	# of Pages	Time to Convert to HTML
TO 1C-130(A)H-2-71JG-00-1	Technical Manual Job Guide, Organizational Maintenance, Power Plant Operating Limits and Checklists	213	10 hours from original MS Word files (formatted and combined to one file)

8.2 SGML TO HTML

For comparison, the same 485 pages listed in Table 8-1 were converted from SGML to HTML. Following the process outlined in Figure 8-2, it took 28 hours for the conversion. However, the HTML results were still not usable with HTML Transit. It is believed that with some customization of the ArborText product suite, using the ArborText Command Language that is built into the products, the conversions would have been less time consuming and more efficient. Costs per page are presented in Table 9-3.

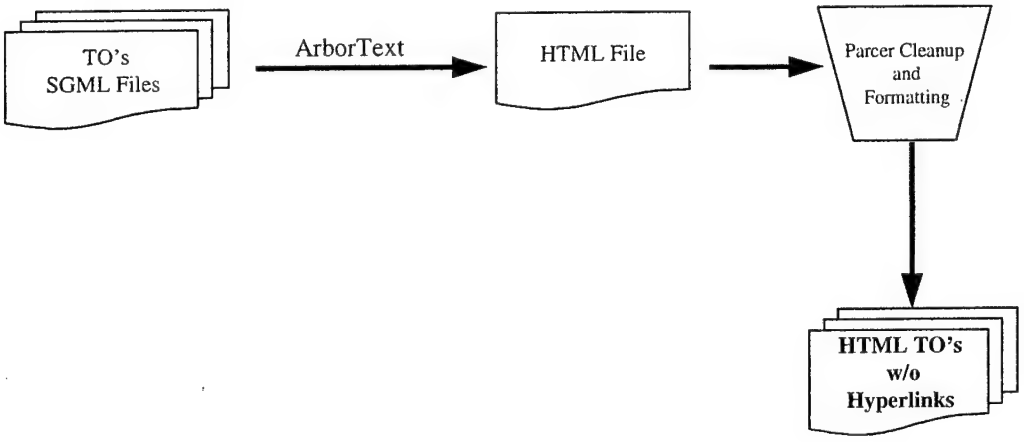


Figure 8-2. SGML to HTML Conversion

9.0 COSTS

Tables 9-1, 9-2, and 9-3 depict costs associated with the conversions completed for the scenario demonstration described in Section 8.0. The conversions related to the scenario were quantified by skill category, labor rate, and the time to convert the paper based information to HTML as demonstrated to Armstrong Lab personnel.

The paper to HTML conversion cost per page ($4.87) using the evaluated COTS software is significantly lower than the paper to IETMs cost per page ($93.55 to $140.74). The team put together the three demonstrations in less than one week.

Table 9-1. Paper to HTML Conversion Costs

Conversion	Task	Skill Level*	Avg. Time/Page	Avg. Cost/Page
Paper to MS Word**	Scan, OCR, Format	Administrative Support I - $43.32/hour	3.49 min.	$2.52
MS Word File***	Format	Administrative Support I - $43.32/hour	2.25 min.	$1.63
MS Word to HTML	Design Templates, JavaScript, Linking	Facility Support Engineer - $43.32/hour	1 min.	$0.72
Total Avg. Time and Cost Per Page			**6.74 min.**	**$4.87**

* Price per hour is based on the TASC GSA ADP Services Schedule
** Based on 485 pages
*** Based on 213 pages

Table 9-2. SGML to HTML Conversion Costs

Conversion	Task	Skill Level	Avg. Time/Page	Avg. Cost/Page
SGML to HTML w/o Links	Verify SGML, Parse, Modify Map File, Save	Principle Member of Technical Staff $60.33/hour	3.6 min.	$3.62
Total Avg. Time and Cost Per Page			**3.6 min.**	**$3.62**

Table 9-3. Paper to IETMs Conversion Costs

Conversion	Task	Skill Level	Avg. Time/Page	Avg. Cost/Page
Paper to Ms Word	Scan, ORC, Format	Administrative Labor Category*	3.49 min.	$2.52
MS Word to SGML	Tag Data	Multiple Labor Categories**	12 min.	$11.09
SGML to IETMS	Analyze, Eliminate, Format, Linking, Traversement	Multiple Labor Categories***	80 – 123 min.	$79.94 to $127.13
Total Avg. Time and Cost Per Page			**95.49 min. - 138.49 min.**	**$93.55 - $140.74**

* Reference Table 9-1. Paper to HTML Conversion Costs
** Reference Table 6-3. Paper to SGML Costs
*** Reference Table 6-5. Cost of IETMs

The costs in Table 9-3 are similar to a 1992 Armstrong Lab Study that cited IETMS cost at $100/page and Lockheed Martin's Automated Conversion System (ACS) process that cited $50 to $100/page (1997).

The following factors have remained constant in the COPIS cost evaluation and could change the conversion costs per page if considered.

- Costs for paper to HTML do not include costs of hardware (i.e., scanners, interface boards, workstations) or software.
- Costs do not reflect creation, re-authoring, or document preparation work (i.e., copying from bound publications).
- All scanned graphics are bitmaps.

10.0 CONFIGURATION MANAGEMENT

In addition to the creation and retrieval of TO information, there is the issue of updating and editing TOs. For the purpose of this study, the concept of multiple versions will be referred to as Configuration Management. It ensures the TOs are managed from a defined baseline to identify, control, and track the status of the TOs so the integrity of the data and the evolution of it is more manageable for the Air Force. It is not to be seen as additional work, but as a systematic way of performing day to day activities. This is done to identify the TOs, control the changes, and track the change level of each individual TO.

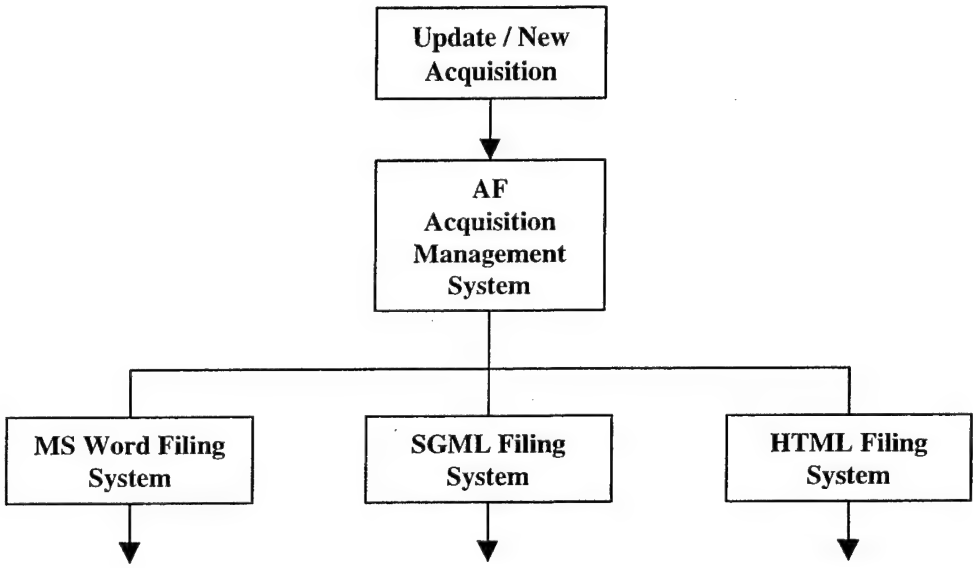


Figure 10-1. Configuration Management

Some basic assumptions were made when researching and analyzing the needs for configuration management. The first assumption is that updates and/or new acquisition data is sent to the OPR. It is also assumed that there will still be a need for paper in some remote areas, and for those areas all files are converted to MS Word using specific templates. The MS Word files will contain certain information that is not viewable to the end user stating 1) which HTML Transit template was used to create the file and 2) the public identifier of the DTD to which the corresponding SGML file is tagged, if applicable. Figure 10-1 shows a top-level look at the configuration management plan.

Since there are some contractors delivering SGML tagged instances, there is a need for further breakdown of the SGML files into chapter fragments to facilitate change packages. The SGML fragments will be converted to HTML for use with a web browser.

The way this configuration management plan will work is highly dependent upon the directory structures of the computer and/or server. The directories will be named by the TO number. Under the TO number there will be directories for the front matter, which includes a file

for the title page, a file for the list of effective pages (also known as the A page), TOC, foreword or preface or introduction, and the safety summary. There will also be a directory for each of the chapter(s), and if need be the chapter can be further broken down by task for system, subsystem, sub subsystem number. And there will be another directory for the rear matter files, which include things like any appendices, indices, or glossaries.

When a TO is updated, only the applicable files need to be replaced, including the title page, A page, and TOC. There will be metadata in the word file telling it which version and type of template was used. There will also be metadata describing which DTD including version number that the file is in conformance with, if applicable. Figure 10-2 shows the MS Word filing system.

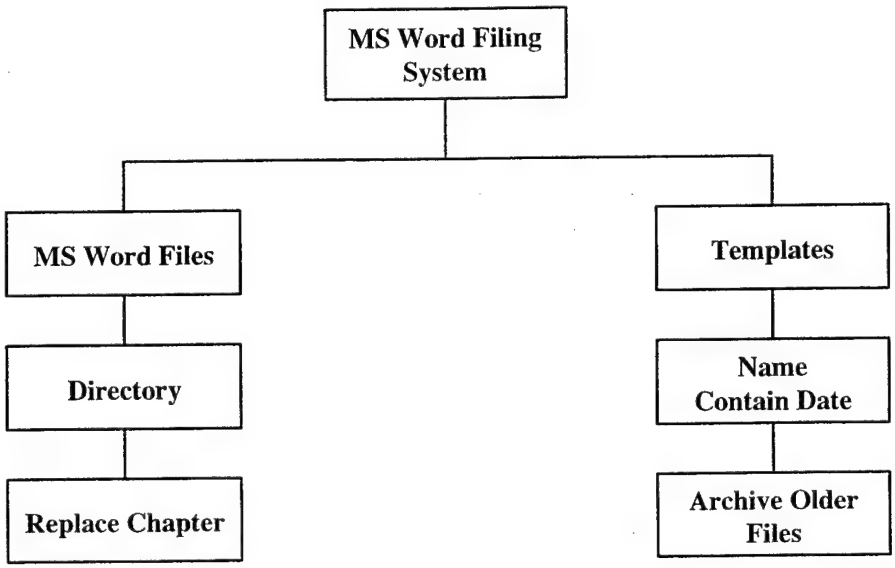


Figure 10-2. MS Word Filing System

If there are applicable SGML files, the directory structure will be the same as for the Word files. For the SGML files, only the applicable fragment will need replacing. If DTD modifications are made, the latest DTD gets a new version number, and the previous version is archived for use with other older files. Figure 10-3 shows the SGML filing system.

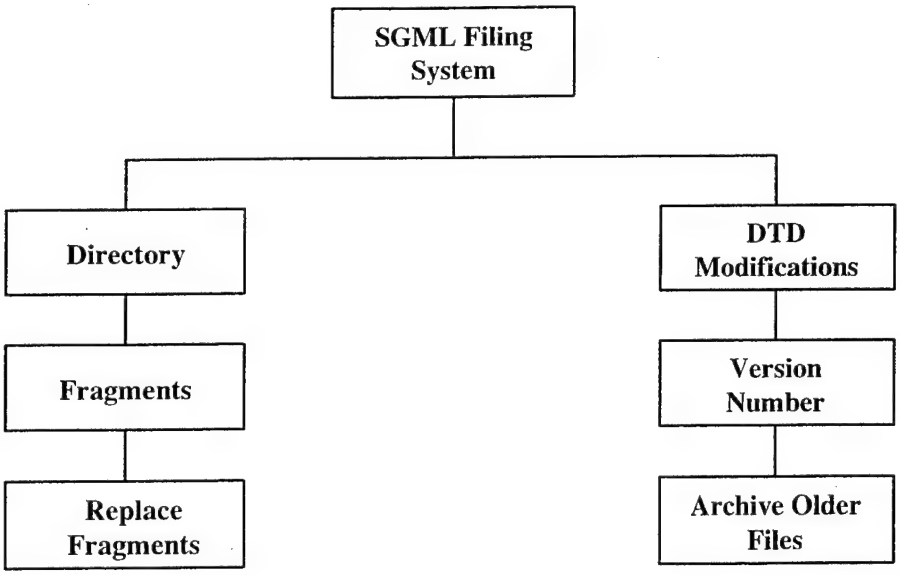


Figure 10-3. SGML Filing System

The HTML files are obtained using HTML Transit. If the HTML Transit templates are modified, a new template file is created which will have as part of its name the current date, and the previous template will be archived.

The HTML files have the same type of directory structure as the MS Word and SGML files, so only the applicable files need to be replaced.

All of these files will be hosted on a web server allowing ample access to the files. Compact Discs (CDs) will be sent to those not having access to the web server every 90 days. The CD will contain an installation script to overwrite the files that have been modified. This will be an automatic process. The user involvement consists of running a simple program, and the computer will overwrite or create the files as applicable.

The AF Distribution Management System is responsible for sending out the CDs and the paper output to the authorized and applicable parties. It is also responsible for the maintenance and support of the web server that contain the TO data.

For those that have access to the web server, every 90 days an alert is sent to the applicable users indicating the need to download the updated files contained on the web server. The user will have the discretion for three days to run an installation script, which will download the newest files to the user's hard drive. On the fourth day the user is alerted that the update installation script was not executed, and an installation script from the web server will automatically update the user's hard drive. This is known as a forced installation. Figure 10-4 shows the HTML filing system along with the distribution system.

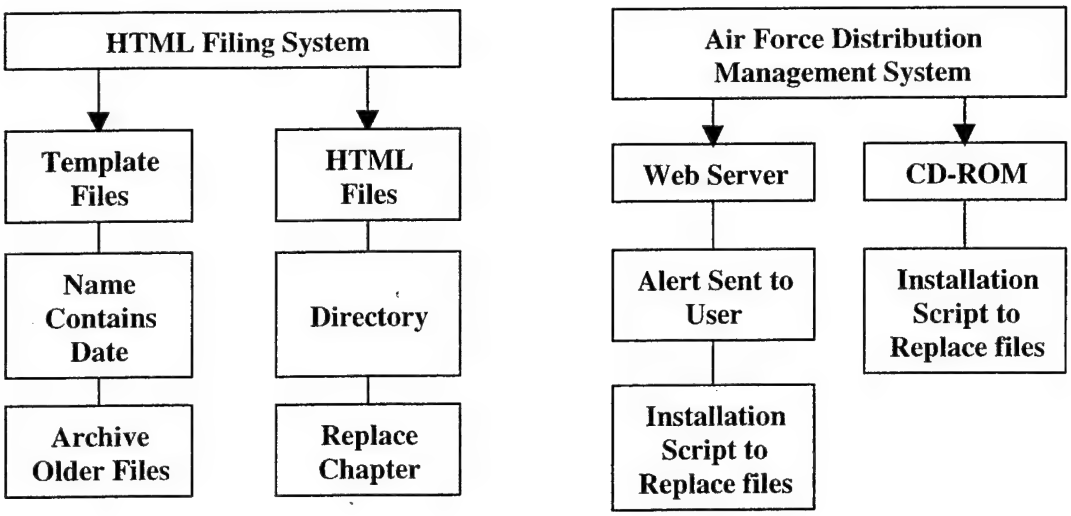


Figure 10-4. HTML Filing System

When the user accepts the download from the web server, a notification is sent back to the AF Distribution Management System alerting the office of that particular user's acceptance or forced installation of the latest updates.

For those remote locations requiring paper or CD media from the AF Distribution Management Office, a form will need to be completed alerting the office of the remote user's acceptance. Once the AF Distribution Management Office has the acceptance replies, it is the user's responsibility to use the most current data. Those receiving the data from the web server will only have access to the most current data. The same applies to the users that run the installation programs on the CDs. The older files are overwritten and therefore not available.

11.0 CONCLUSIONS / RECOMMENDATIONS

This study has shown the feasibility of using low-end COTS software packages to convert paper-based technical information to usable HTML files for significantly less than the cost to convert to IETMs. The information converted to non-proprietary HTML can be viewed in low cost web browsers such as MS Internet Explorer and Netscape Navigator. In addition to cost savings, by using the HTML format, the USAF will be able to take advantage of open market and commercially competitive product upgrades for HTML and future generation file formats. There are some minor risks associated with the process described in the COPIS report. They are discussed in Section 11.1. Further areas of study related to the conversion process are discussed in Section 11.2.

The process in Figure 8-1 provides for changes to software resulting in improved HTML conversion translations and output. Due to the current rapid evolution of Document Management/Web Publishing Technologies, upgrades to COTS software in the following areas are likely:

- OCR Software
- Web Browsers
- HTML Translation (with linking & pattern matching capabilities).

11.1 RISKS – MITIGATION

Table 11-1 lists the risks associated with the current process. Potential mitigation to these risks are also presented.

Table 11-1. Risks & Mitigation

Risk	Mitigation
Cost - Resources to Cleanup & Format from OCR could outweigh rekeying	Training, manual intervention, institute quality checks
Missed links across TOs – especially in Configuration Management	Manual intervention, institute quality checks, added functionality to translation packages
Rapid evolution of Document Management Technologies	Thorough evaluations of current technology and current events

11.2 FURTHER AREAS OF STUDY

In this section, areas for further study are identified. The team identified the following four areas for further study based on lessons learned throughout the study. The first three areas for further study addresses technologies and areas, which the team believes could be exploited to better the process of converting to HTML. The final area addressed, XML, is a fairly new DTD, which could ultimately prove to be a more dominant and adopted DTD than HTML.

11.2.1 Single Use of Graphic File for Multiple Display

In the process of converting a paper document to HTML pages, identical graphics were created multiple times. This is a result of the scanning and OCR process. Assume that we are converting a paper document that contains the same diagram five times throughout the document. When this document is scanned and then run through an OCR package, the resulting word processing file will have five separate copies of the image in the document. The reason for the multiple copies is that the OCR package does not try and determine if there are identical images from the scanned document. Using imaging technology, it could possibly be feasible during the OCR process to identify graphics that were identical and to remove all redundant copies and replace them with references to the single copy. Image comparison algorithms could be used to determine within a certain degree of accuracy whether two images were identical. The team recommends that further research should be conducted to investigate imaging packages, which could analyze the document in the OCR phase of the conversion to identify redundant graphics. This would be beneficial in the overall process of converting to HTML. One advantage of removing redundant copies of graphics throughout the HTML files is that if a change needs to be made to a particular graphic, it needs only to be changed one time. Another advantage would be the reduction in the amount of disk space needed to store the converted TOs.

11.2.2 Program for Automatic Styling

One of the major problems with the MS Word file produced from the OCR process was that it lacked paragraph styles. As discussed in the details of HTML Transit, character and paragraph styles are what HTML Transit keys from for creating Transit Elements. In this study, the team manually formatted the document produced from the OCR process. This manual processing consisted of identifying chapters and sections in the document and applying the appropriate style. The team recommends that further research should be conducted in order to determine the feasibility of creating a program that automatically parses through the document and inserts the appropriate style. The generation of such a program could greatly reduce the amount of time needed to translate an existing paper TO.

11.2.3 Pattern Matching

One of the powerful features HTML Transit exhibited was its ability to identify patterns in the source document. The team attempted to exploit this capability in order to automatically generate hyperlinks between different TOs. Unfortunately, it was discovered that the pattern

matching capability available in HTML Transit was not robust enough to handle this issue. The main feature lacking in the pattern matching capability found in HTML Transit was that there was no way to manipulate the identified pattern once it was found in the document. This functionality was needed in order to construct a link to a different TO. Potential solutions include HTML Transit adding additional functionality to its pattern matching capability, as well as researching the development of a program, which could automatically insert the links to separate TOs.

11.2.4 XML

XML is a restricted form of SGML (ISO 8879). It was created and developed by the W3C XML Working Group. The goal of XML is to enable generic SGML to be served, received, and processed on the WWW in the way that is now possible with HTML. It has been designed for ease of implementation and for interoperability with both SGML and HTML. XML was issued as a recommended practice by the W3C on 8 December 1997. This means it is ready for review and voting to become a W3C recommendation.

Much like SGML, the basic features of XML include vendor independence, user extensibility, complex structures, validation and human readability. Currently, some existing commercial tools such as the ArborText product suite and a number of freeware products can process it.

XML has been called a "bridge" between SGML and HTML. It is primarily intended to meet the requirements of large-scale Web content providers for industry-specific markup, vendor-neutral data exchange, media-independent publishing, one-on-one marketing, workflow management in collaborative authoring environments, and the processing of Web documents by intelligent clients. XML can support the Unicode character set in both its UTF-8 and UTF-16 encoding. It is designed for the quickest possible client-side processing consistent with its primary purpose as an electronic publishing and data interchange format.

The W3C XML Working Group includes key industry players such as Adobe, ArborText, DataChannel, Grif, Inso, Hewlett-Packard, Isogen, Microsoft, NCSA, Netscape, SoftQuad, Sun Microsystems, and Fuji Xerox; as well as experts in structured documents and electronic publishing.

The XML1.0 specification has been produced as part of the W3C XML Activity, and is available at http://www.w3.org/TR/PR-xml-971208. For more information on XML, please see http://www.w3.org/XML/

ACRONYMS

ACD - Adobe Capture Document

ACS - Automated Conversion System (a conversion to IETM service provided Lockheed Martin)

AF PDSM - Air Force Product Data Systems Modernization

AIMSS - Advanced Integrated Maintenance Support System

AFRL/HESR – Air Force Research Laboratory/Human Effectiveness Systems Division, Readiness Branch

CALS - Continuous Acquisition Lifecycle Support

CCITT - Consultative Committee on International Telegraphy and Telephony

COPIS - Conversion of Paper-Based Information Study

COTS - Commercial Off The Shelf

DOC - Word Document File Extension

DOD - Department of Defense

DPI - Dots Per Inch

DTD - Document Type Definition

DOC - Word Document File Extension

HTML - HyperText Markup Language

HTSC - Hughes Technical Services Company

IETM - Interactive Electronic Technical Manual

MID - Metafile for Interactive Documents

MIL-PRF-87269 (MIL-D-87269) - Military Performance Standard, "Database, Revisable: Interactive Electronic Technical Manuals, For the Support Of"

MIL-PRF-87268 - "Manuals, Interactive Electronic Technical: General Content, Style, Format, and User-Interaction Requirements"

MS - Microsoft

OCR/ICR – Optical Character Recognition/Intelligent Character Recognition

PC – Personal Computer

PDF – Portable Document Format (Adobe proprietary)

RTF – Rich Text Format

SGML – Standard Generalized Markup Language

OPR – Office of Primary Responsibility

TAMSCO – Technical And Management Services Corporation

TIFF – Tagged Image File Format

TO – Technical Order

TOC – Table of Contents

TODO – Technical Order Distribution Office

USAF – United States Air Force

WC3 – World Wide Web Consortium

WWW – World Wide Web

XML – Extensible Markup Language